

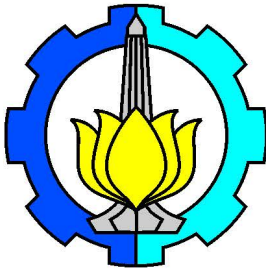
TESIS - SM 142501

**MODEL RANTAI PASOK MENGGUNAKAN *PETRI NET*  
DAN ALJABAR *MAX PLUS* DENGAN  
MEMPERTIMBANGKAN PRIORITAS KAPAL TANKER**

SHOFIYATUL MUFIDAH  
1213 201 034

DOSEN PEMBIMBING  
Dr. Subiono, M.S.

PROGRAM MAGISTER  
JURUSAN MATEMATIKA  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
INSTITUT TEKNOLOGI SEPULUH NOPEMBER  
SURABAYA  
2015



THESIS - SM 142501

# **SUPPLY CHAIN MODEL USING PETRI NET AND MAX PLUS ALGEBRA BY CONSIDERING TANKER SHIP PRIORITY**

SHOFIYATUL MUFIDAH  
1213 201 034

SUPERVISOR:  
Dr. Subiono, M.S.

MAGISTER PROGRAM  
DEPARTMENT OF MATHEMATICS  
FACULTY OF MATHEMATICS AND NATURAL SCIENCE  
INSTITUT TEKNOLOGI SEPULUH NOPEMBER  
SURABAYA  
2015



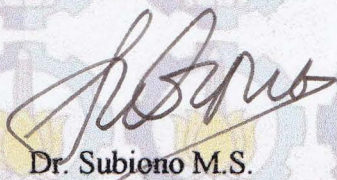
MODEL RANTAI PASOK MENGGUNAKAN *PETRI NET* DAN ALJABAR  
*MAX PLUS* DENGAN MEMPERTIMBANGKAN PRIORITAS KAPAL  
TANKER

Telah disusun untuk memenuhi salah satu syarat memperoleh gelar  
Magister Sains (M.Si)  
di  
Institut Teknologi Sepuluh Nopember

oleh:  
Shofiyatul Mufidah  
Nrp. 1213 201 034

Tanggal Ujian : 12 Maret 2015  
Periode Wisuda : September 2015

Disetujui oleh:



Dr. Subiono M.S.  
NIP: 19570411 198403 1 001

(Pembimbing)



Dr. Hariyanto, M.Si.  
NIP: 19530414 198203 1 002

(Penguji)



Prof. Dr. Erna Apriliani, M.Si.  
NIP: 19660414 199102 2 001

(Penguji)



Direktur Program Pascasarjana,

Prof. Dr. Ir. Adi Soeprijanto, M.T.  
NIP. 19640405 199002 1 001



## **Model Rantai Pasok Menggunakan *Petri Net* dan Aljabar *Max Plus* dengan Mempertimbangkan Prioritas Kapal Tanker**

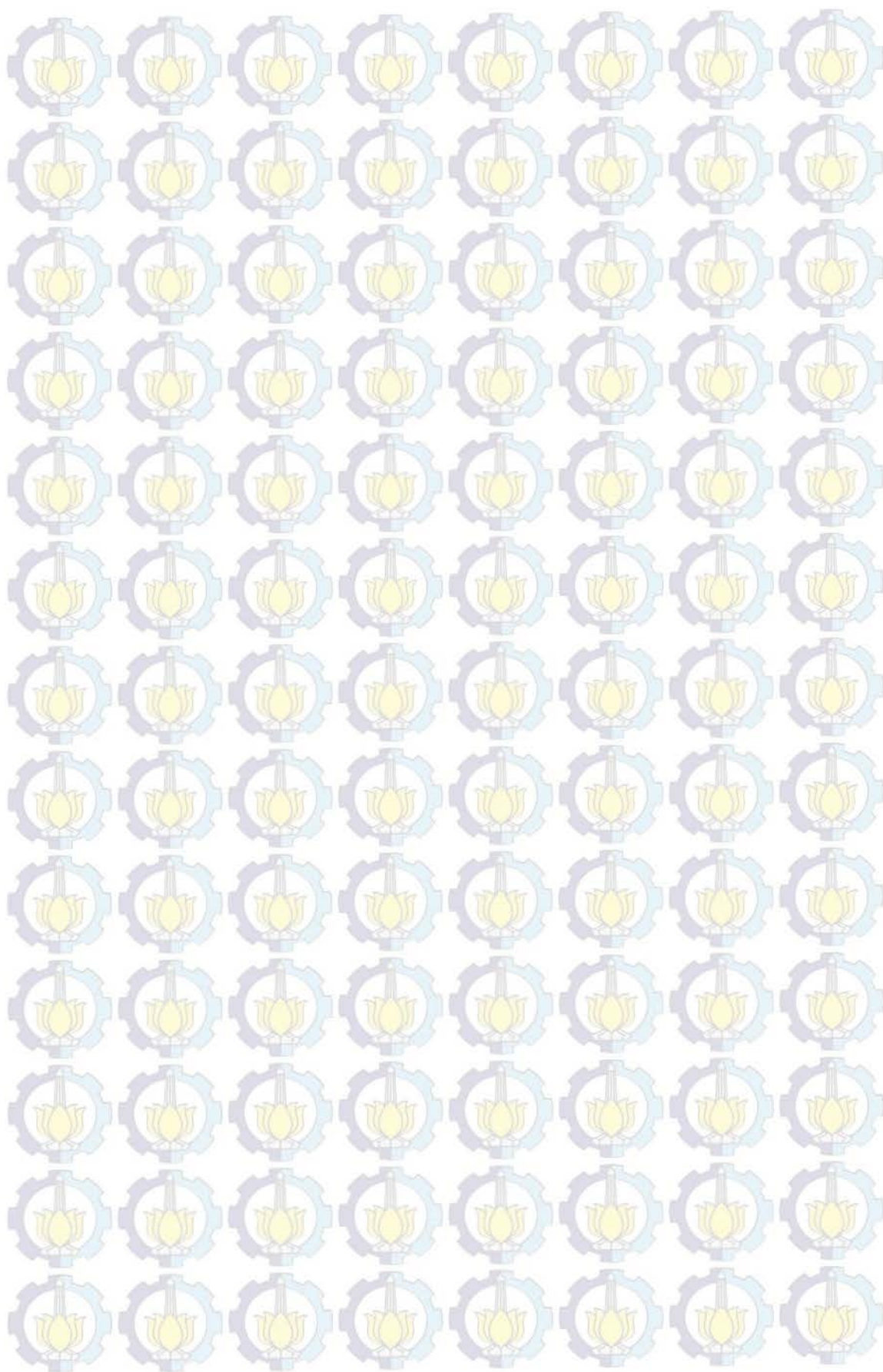
Nama : Shofiyatul Mufidah  
NRP : 1213201034  
Dosen Pembimbing : Dr. Subiono, M.S.

### **ABSTRAK**

Salah satu aplikasi *Petri Net* dan Aljabar *Max Plus* adalah untuk membuat model rantai pasok pada proses distribusi hasil produksi dari *supplier* menuju *customer*. Berdasarkan model rantai pasok yang telah dibuat dapat dirancang jadwal pengiriman hasil produksi, sehingga permintaan *customer* bisa dipenuhi tepat waktu oleh *supplier*. Pada penelitian ini dibuat model rantai pasok pada proses pengiriman solar milik Pertamina yaitu dari Terminal Bahan Bakar Minyak (TBBM) Tuban sebagai *supplier* menuju TBBM Manggis di Bali dan Tanjung Wangi di Banyuwangi sebagai *customer*. Proses pengiriman menggunakan 5 kapal tanker (3 kapal tanker berkapasitas 22.000 Kilo Liter (Kl) dan 2 kapal tanker lainnya berkapasitas 25.000 Kl). Rantai pasok dimodelkan menggunakan *Petri Net* dengan mempertimbangkan prioritas kapal tanker. Pada model pertama kapasitas kapal tanker diasumsikan sama (22.000 Kl) dan pengiriman diprioritaskan menggunakan kapal tanker yang telah kembali dari pengiriman sebelumnya. Sedangkan pada model kedua diasumsikan volume permintaan sampai 25.000 Kl dan pengiriman untuk volume tidak melebihi 22.000 Kl diprioritaskan menggunakan kapal tanker dengan kapasitas 22.000 Kl. Kemudian model *Petri Net* dijabarkan dalam bentuk Aljabar *Max Plus*. Selanjutnya disusun jadwal mulai *loading* dan keberangkatan kapal tanker. Jadwal dibuat dalam program Scilab dengan menggunakan data yang telah dikumpulkan. Pada model kedua permintaan diasumsikan sebanyak 26 seperti pada model pertama namun 11 diantaranya dengan volume 22.000 Kl-25.000 Kl. Berdasarkan hasil simulasi, diperoleh jadwal *loading* dan keberangkatan dari kapal tanker. Pada model pertama, dengan pengaturan prioritas kapal tanker maka 26 permintaan dari *customer* dapat dipenuhi tepat waktu hanya dengan menggunakan 4 kapal tanker. Sedangkan pada model kedua, dengan pengaturan prioritas kapal tanker maka 26 permintaan dengan volume sampai 25.000 Kl dapat dipenuhi tepat waktu menggunakan 5 kapal tanker.

Kata Kunci : *Petri Net*, Aljabar *Max Plus*, rantai pasok, kapasitas kapal tanker, Terminal Bahan Bakar Minyak (TBBM)





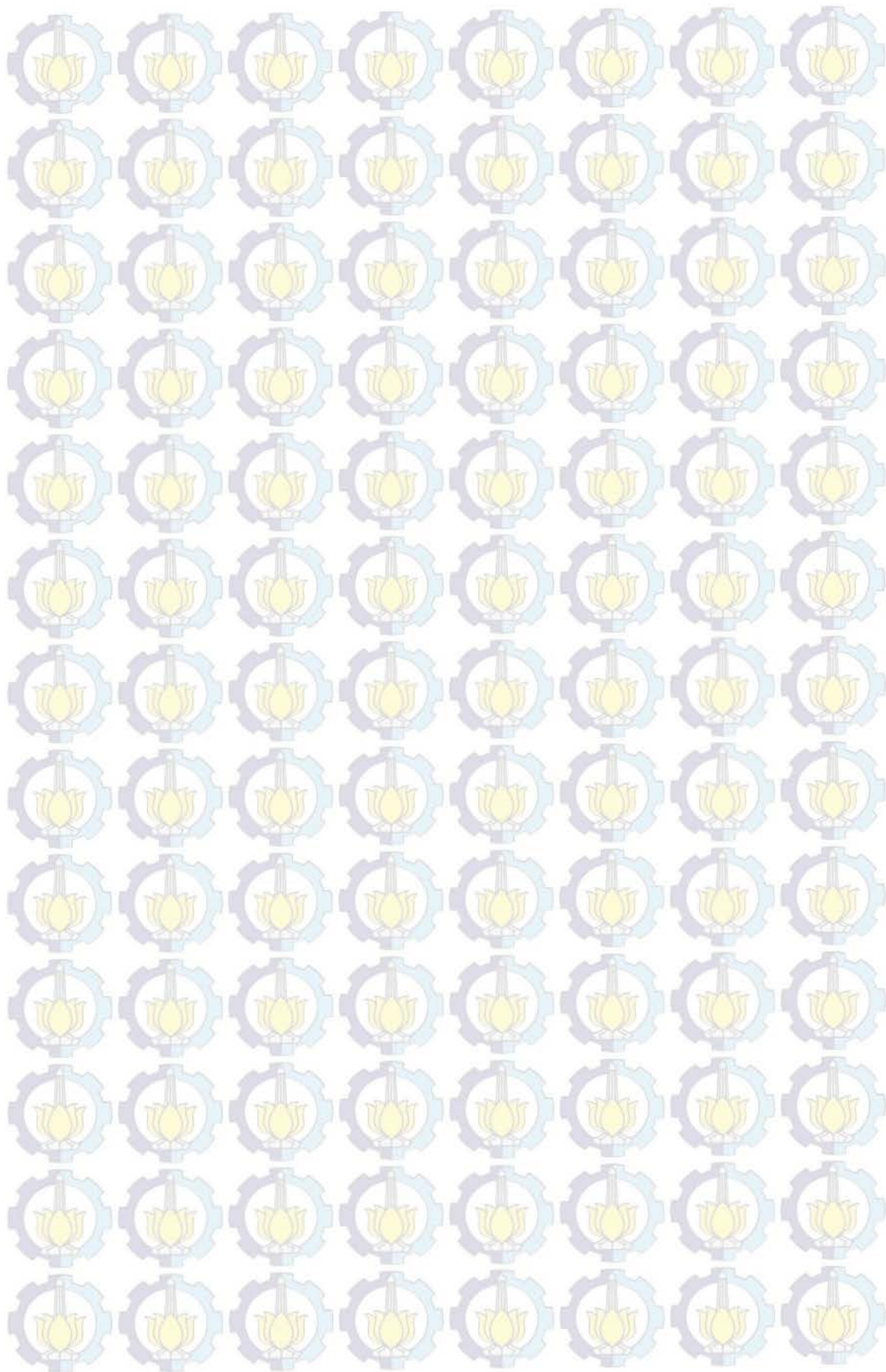
# SUPPLY CHAIN MODEL USING PETRI NET AND MAX PLUS ALGEBRA BY CONSIDERING TANKER SHIP PRIORITY

Name : Shofiyatul Mufidah  
NRP : 1213201034  
Supervisor : Dr. Subiono, M.S.

## ABSTRACT

One of Petri Net and Max Plus Algebra application is to create a supply chain model of distribution process from supplier to customer. Based on the supply chain model that has been created, it can designed the delivery schedule of production result so that the customer demand can be met on time by the supplier. In this research made supply chain model of diesel fuel delivery in Pertamina from Terminal Bahan Bakar Minyak (TBBM) Tuban as supplier to TBBM Manggis in Bali and Tanjung Wangi in Banyuwangi as customer. Delivery processing using 5 tanker ships (the capacities of 3 tanker ships is 22.000 Kilo Liter (Kl) and the other tanker ships is 25.000 Kl). Supply chain is modeled using Petri Net by considering tanker ship priority. In the first model assumed the capacities of tanker ships is equal (22.000 Kl) and delivery prioritized using tanker ship that has been returned from a previous delivery. While in the second model assumed that the volume of demand until 25.000 Kl and delivery for the demand with volume does not exceed 22.000 Kl prioritized using tanker ships with capacity 22.000 Kl. And then Petri Net model is translated into Max Plus Algebra model. And then arranged the loading start and departure schedules of tanker ships. The schedule is made in Scilab program using data that has been collected. In the second model, there are assumed 26 demands as in the first model, but the volume of 11 of them is 22.000 Kl-25.000 Kl. Based on simulation results, obtained start loading and departure schedule of tanker ships. In the first model, with priority setting of tanker ships make the 26 demands from customer can met on time only by using 4 tanker ships. While in the second model, with priority setting of tanker ships make 26 demands with volume until 25.000 Kl can met on time using 5 tanker ships.

Key word: Petri Net, Max Plus Algebra, supply chain, capacity of tanker ship, Terminal Bahan Bakar Minyak (TBBM).





## KATA PENGANTAR

Puji syukur kehadiran kepada Allah SWT atas limpahan rahmat, taufik, hidayah, serta inayah-Nya sehingga penulis dapat menyelesaikan tesis dengan judul **“MODEL RANTAI PASOK MENGGUNAKAN *PETRI NET* DAN ALJABAR *MAX PLUS* PADA RANTAI PASOK DENGAN MEMPERTIMBANGKAN PRIORITAS KAPAL TANKER”** ini. Sholawat salam senantiasa tercurahkan kepada Baginda Rosulullah SAW.

Tesis ini disusun untuk memenuhi salah satu persyaratan memperoleh gelar Magister Sains (M.Si) di Institut Teknologi Sepuluh Nopember (ITS) Surabaya. Penulis menyadari bahwa terselesaikannya tesis ini tidak lepas dari bantuan yang sangat berarti dari banyak pihak. Oleh karena itu, pada kesempatan ini penulis menyampaikan ucapan terima kasih yang tak terhingga kepada :

1. Ibu Prof. Dr. Erna Apriliani selaku Ketua Jurusan Matematika FMIPA ITS sekaligus menjadi dosen wali dari penulis selama pendidikan Pra Magister serta menjadi dosen penguji tesis yang senantiasa memberikan motivasi, bimbingan dan arahan selama menempuh pendidikan di ITS.
2. Bapak Dr. Subiono, M.S. selaku Ketua Program Studi Pascasarjana Matematika ITS sekaligus sebagai dosen pembimbing yang dengan sabar dan penuh pengertian serta ikhlas meluangkan waktu, tenaga, dan pikiran untuk membimbing, memotivasi dan memberikan saran-saran yang sangat berharga kepada penulis selama penyusunan tesis ini.
3. Ibu Dr. Dwi Ratna Sulistyaningrum, S.Si., M.T. selaku dosen wali dari penulis yang telah banyak memotivasi dan memberikan saran-saran yang sangat bermanfaat.

4. Seluruh dosen matematika ITS yang telah memberikan ilmu yang sangat berharga serta staf kependidikan jurusan matematika yang telah banyak membantu penulis.
5. Bapak Drs. Djoko Adi Susilo, M.Pd dan Bapak Drs. I Ketut Suastika, M.Pd yang telah banyak membantu penulis sehingga bisa melanjutkan kuliah di Program Pasca Sarjana ITS melalui jalur beasiswa.
6. Pak Fahim dan Pohet yang sudah banyak mengajari dalam pembuatan model dan program.
7. Mbak Widdya Putri Sierliawati yang telah banyak membantu dalam pengumpulan data-data sekunder penelitian.
8. Kedua orang tuaku (Bapak M. Anwari dan Ibu Tutik), kedua adikku (Mardiyatul Husnah dan Karomatuz Zuhriyah) serta seluruh keluarga yang tak pernah berhenti mendo'akan dan memberikan perhatian.
9. Tunanganku (mas Dian) yang telah sabar menunggu dan senantiasa memotivasi serta keluarganya yang tak pernah berhenti mendoakan kelancaran studi.
10. Semua pihak yang membantu terselesaikannya tesis ini yang tidak dapat disebutkan satu persatu.

Penulis menyadari bahwa di dalam tesis ini masih terdapat banyak kekurangan yang disebabkan oleh keterbatasan pengetahuan dan pengalaman penulis. Oleh karena itu, penulis mengharapkan kritik dan saran yang membangun dari para pembaca demi kesempurnaan tesis ini.

Akhir kata, semoga tesis ini dapat bermanfaat bagi semua pihak, Amin.

Surabaya, 24 Maret 2015

Penulis

## DAFTAR ISI

<b>LEMBAR PENGESAHAN . . . . .</b>	<b>i</b>
<b>ABSTRAK . . . . .</b>	<b>iii</b>
<b>ABSTRACT . . . . .</b>	<b>v</b>
<b>KATA PENGANTAR. . . . .</b>	<b>vii</b>
<b>DAFTAR ISI . . . . .</b>	<b>ix</b>
<b>DAFTAR GAMBAR . . . . .</b>	<b>xi</b>
<b>BAB 1 PENDAHULUAN . . . . .</b>	<b>1</b>
1.1 Latar Belakang . . . . .	1
1.2 Perumusan Masalah . . . . .	3
1.3 Batasan Masalah . . . . .	4
1.4 Tujuan Penelitian. . . . .	4
1.5 Manfaat Penelitian . . . . .	5
<b>BAB 2 KAJIAN PUSTAKA DAN DASAR TEORI . . . . .</b>	<b>7</b>
2.1 Penelitian-Penelitian sebelumnya . . . . .	7
2.2 Definisi Solar . . . . .	8
2.3 Definisi Rantai Pasok. . . . .	8
2.4 Definisi Terminal Bahan Bakar Minyak (TBBM) . . . . .	9
2.5 Wilayah Kerja Pertamina <i>Supply and Distribution Region III</i> . .	9
2.6 Aljabar <i>Max Plus</i> . . . . .	10
2.7 <i>Petri Net</i> . . . . .	13
<b>BAB 3 METODE PENELITIAN. . . . .</b>	<b>17</b>
<b>BAB 4 HASIL DAN PEMBAHASAN . . . . .</b>	<b>21</b>
4.1 Model Rantai Pasok menggunakan <i>Petri Net</i> dengan Kapasitas Kapal Tanker dianggap sama. . . . .	21
4.2 Model Aljabar <i>Max Plus</i> dari <i>Petri Net</i> dengan Kapasitas Kapal Tanker dianggap sama . . . . .	31
4.3 Data permintaan yang digunakan dalam Simulasi Program . . .	38
4.4 Penjadwalan Model pertama dengan Program Scilab . . . . .	39



4.5 Model Rantai Pasok menggunakan <i>Petri Net</i> dengan mempertimbangkan perbedaan kapasitas Kapal Tanker. . . . .	45
4.6 Model Aljabar <i>Max Plus</i> dari <i>Petri Net</i> dengan mempertimbangkan perbedaan kapasitas Kapal Tanker . . . . .	56
4.7 Data-Data sekunder untuk Model kedua yang digunakan dalam Simulasi Program. . . . .	65
4.8 Penjadwalan Model kedua dengan Program Scilab. . . . .	65
<b>BAB 5 KESIMPULAN.</b> . . . .	71
5.1 Kesimpulan . . . . .	71
5.2 Saran . . . . .	72
<b>DAFTAR PUSTAKA.</b> . . . .	75
<b>LAMPIRAN-LAMPIRAN</b> . . . . .	77
Lampiran 1.A. Data Volume dan Waktu Permintaan untuk Model pertama. . . . .	77
Lampiran 1.B. Data yang disimpan dalam File Excel untuk Model pertama . . . . .	78
Lampiran 2.A. Listing Program Fungsi date2intetc. . . . .	81
Lampiran 2.B. Listing Program Fungsi int2dateetc. . . . .	82
Lampiran 3.A. Listing Program Fungsi date2stretc. . . . .	83
Lampiran 3.B. Listing Program Fungsi supplychain1to2 . . . . .	84
Lampiran 4.A Data Volume dan Waktu Permintaan untuk Model kedua. . . . .	89
Lampiran 4.B. Data yang disimpan dalam File Excel untuk Model kedua. . . . .	90
Lampiran 5.A Listing Program Fungsi Supplychain1to2b. . . . .	97

## DAFTAR GAMBAR

Gambar 2.1	Peta Wilayah Kerja PT. Pertamina <i>Supply and Distribution Region III</i> . . . . .	9
Gambar 2.2	Contoh <i>Petri Net</i> sederhana . . . . .	14
Gambar 4.1	Keadaan awal Model <i>Petri Net</i> dengan Kapasitas Kapal Tanker dianggap sama . . . . .	23
Gambar 4.2	Keadaan setelah Transisi $T_{10}$ <i>Difire</i> sebanyak 1 Kali . . . . .	25
Gambar 4.3	Keadaan setelah Transisi $T_0$ <i>Difire</i> sebanyak 1 Kali . . . . .	26
Gambar 4.4	Keadaan setelah Transisi $T_1$ <i>Difire</i> menyebabkan Transisi $T_2$ dan $T_3$ menjadi <i>enabled</i> . . . . .	26
Gambar 4.5	Keadaan setelah Transisi $T_2$ <i>Difire</i> menunjukkan keberangkatan Kapal Tanker menuju TBBM Manggis. . . . .	27
Gambar 4.6	Keadaan setelah Transisi $T_3$ <i>Difire</i> menunjukkan keberangkatan Kapal Tanker menuju TBBM Tanjung Wangi. . . . .	28
Gambar 4.7	Keadaan setelah Transisi $T_5$ <i>Difire</i> . . . . .	29
Gambar 4.8	Keadaan setelah Transisi $T_7$ <i>Difire</i> menyebabkan $T_0$ tidak <i>Enabled</i> . . . . .	30
Gambar 4.9	Keadaan setelah Transisi $T_6$ <i>Difire</i> menyebabkan $T_0$ tidak <i>Enabled</i> . . . . .	30
Gambar 4.10	Hasil Simulasi Program berupa Penjadwalan untuk permintaan ke-1 sampai ke-16. . . . .	43
Gambar 4.11	Hasil Simulasi Program berupa penjadwalan untuk Permintaan ke-11 sampai ke-26. . . . .	43
Gambar 4.12	Keadaan awal <i>Petri Net</i> untuk Model Rantai Pasok dengan Volume sampai 25.000 Kl . . . . .	46
Gambar 4.13	Keadaan setelah Transisi $T_{16}$ dan $T_{17}$ <i>Difire</i> sebanyak 1 Kali . . . . .	49
Gambar 4.14	Keadaan setelah Transisi $T_{21}$ dan $T_{22}$ <i>Difire</i> sebanyak 1 Kali . . . . .	50

Gambar 4.15	Keadaan setelah Transisi $T_{18}$ dan $T_{19}$ <i>Difire</i> . . . . .	51
Gambar 4.16	Keadaan setelah Transisi $T_1$ dan $T_2$ <i>Difire</i> . . . . .	52
Gambar 4.17	Keadaan setelah Transisi $T_5$ dan $T_6$ <i>Difire</i> . . . . .	53
Gambar 4.18	Keadaan setelah Transisi $T_9$ dan $T_{10}$ <i>Difire</i> . . . . .	54
Gambar 4.19	Keadaan setelah Transisi $T_{12}$ dan $T_{13}$ <i>Difire</i> . . . . .	54
Gambar 4.20	Keadaan setelah Transisi $T_{14}$ dan $T_{15}$ <i>Difire</i> . . . . .	55
Gambar 4.21	Keadaan ketika Transisi $T_{23}$ Menjadi <i>Enabled</i> . . . . .	56
Gambar 4.22	Hasil Simulasi Program Berupa Penjadwalan Model kedua untuk Permintaan ke-1 sampai ke-6 . . . . .	69
Gambar 4.23	Hasil Simulasi Program berupa Penjadwalan Model kedua untuk Permintaan ke-11 sampai ke-26 . . . . .	69
	.	



## BIOGRAFI PENULIS



Shofiyatul Mufidah, lahir pada 9 Maret 1990 di Pasuruan. Penulis merupakan anak pertama dari tiga bersaudara pasangan Bapak M. Anwari dan Ibu Tutik. Penulis telah menempuh pendidikan formal dari jenjang Pendidikan Dasar di Madrasah Ibtida'iyah (MI) Miftahul Ulum Andonosari pada tahun 1996-2002, kemudian jenjang SLTP di Madrasah Tsanawiyah (MTs) Ya-Ikhsan Andonosari pada tahun 2002-2005 dan dilanjutkan jenjang SLTA di Madrasah Aliyah (MA) Darut Taqwa Sengonagung Purwosari pada tahun 2005-2008. Setelah itu, penulis menempuh pendidikan S1 di Jurusan Pendidikan Matematika Universitas

Kanjuruhan Malang pada tahun 2008-2012. Kemudian penulis melanjutkan pendidikan melalui jalur beasiswa di Pascasarjana Matematika Institut Teknologi Sepuluh Nopember (ITS) Surabaya pada tahun 2012-2014 dengan pendidikan Pra Magister selama satu tahun yakni pada tahun 2012-2013 dan dilanjutkan dengan pendidikan Magister mulai tahun 2013 hingga 2015 dengan NRP. 1213 201 034. Adapun kritik, saran ataupun pertanyaan mengenai Tesis ini dapat menghubungi penulis via email [Sofiezu.SZ@gmail.com](mailto:Sofiezu.SZ@gmail.com).

# **BAB 1**

## **PENDAHULUAN**

Pada bab ini dibahas latar belakang penelitian beserta rumusan permasalahan yang akan dikaji. Berdasarkan rumusan permasalahan tersebut kemudian dibuat batasan-batasan masalah yang digunakan untuk membatasi ruang lingkup penelitian. Selanjutnya dituliskan tujuan dan manfaat yang diharapkan dari kegiatan penelitian.

### **1.1 Latar Belakang**

Di era modern ini kebutuhan Bahan Bakar Minyak (BBM) semakin tinggi seiring dengan pertumbuhan industri dan transportasi. Salah satu jenis BBM yang dikenal di Indonesia adalah solar. Solar merupakan salah satu jenis bahan bakar yang digunakan secara luas untuk keperluan industri dan transportasi antara lain sebagai bahan bakar untuk mesin diesel pada kendaraan bermotor seperti bus, truk dan traktor. Solar merupakan salah satu hasil produksi dari sebuah Badan Usaha Milik Negara (BUMN) yang bertugas mengelola penambangan minyak dan gas bumi di Indonesia yaitu Perseroan Terbatas (PT.) Pertamina (Persero).

Setiap perusahaan manufaktur memiliki paling tidak satu jaringan rantai pasok (*supply chain*). Kegiatan yang termasuk ke dalam jaringan rantai pasok adalah proses distribusi atau pengiriman hasil produksi pada *customer* melalui sistem distribusi. Salah satu kegiatan rantai pasok pada PT. Pertamina adalah proses distribusi produk BBM jenis solar dari Terminal Bahan Bakar Minyak (TBBM) yang menjadi *supplier* menuju TBBM-TBBM lain yang menjadi *customer*. TBBM-TBBM tersebut menerima pasokan solar dari TBBM *supplier* melalui jalur darat maupun laut. Pasokan dari jalur laut dikirim menggunakan kapal tanker khusus milik Pertamina.

Untuk menunjang keberhasilan proses distribusi pada perusahaan manufaktur seperti PT. Pertamina diperlukan pengelolaan rantai pasok (*supply chain management*) yang tepat. Salah satu teknik pengelolaan rantai pasok pada distribusi solar di jalur laut adalah pembuatan jadwal *loading* dan keberangkatan

kapal tanker. Jadwal tersebut perlu dibuat oleh PT. Pertamina agar pengiriman pasokan solar sesuai dengan waktu permintaan yang telah ditetapkan oleh TBBM *customer* sehingga tidak terjadi keterlambatan ataupun kedatangan yang lebih awal. Keterlambatan pengiriman bisa menyebabkan terjadinya kelangkaan solar di sejumlah Stasiun Pengisian Bahan Bakar Umum (SPBU). Sedangkan kedatangan lebih awal bisa menyebabkan terjadinya antrian kapal tanker di dermaga. Selain itu kedatangan lebih awal juga bisa menyebabkan terjadinya penumpukan solar yaitu keadaan ketika kapasitas tempat penyimpanan solar di TBBM *customer* tidak mencukupi untuk diisi pasokan solar lagi.

Salah satu aplikasi *Petri Net* dan Aljabar *Max Plus* adalah untuk merancang penjadwalan. Adzkieya (2008) membuat model *Petri Net* untuk merancang penjadwalan nyala lampu lalu lintas di persimpangan sehingga dapat memberikan kepastian waktu tunggu bagi para pengguna jalan, mengurangi kemacetan dan tingkat kecelakaan serta dapat memperlancar arus kendaraan. Sebelumnya Elmahi dkk. (2003) menggunakan model Aljabar *Max Plus* untuk merancang penjadwalan rantai pasok dari satu *supplier* menuju satu *customer* tanpa memperhatikan waktu *loading* dan *unloading* produk. Selanjutnya Sierliawati (2014) menggunakan *Petri Net* untuk membuat model rantai pasok distribusi solar dari 1 *supplier* yaitu TBBM Tuban menuju 2 *customer* yaitu TBBM Manggis dan Tanjung Wangi menggunakan 5 kapal tanker dengan memperhatikan waktu *loading* dan *unloading* produk namun tanpa memperhatikan prioritas kapal tanker.

Berdasarkan latar belakang tersebut maka pada penelitian ini dibuat model rantai pasok distribusi solar menggunakan *Petri Net* dan Aljabar *Max Plus* dengan mempertimbangkan prioritas kapal tanker. Rantai pasok pada penelitian ini adalah distribusi solar dari TBBM Tuban menuju 2 *customer* yaitu TBBM Manggis dan Tanjung Wangi menggunakan 5 kapal tanker (3 kapal tanker berkapasitas 22.000 Kl dan 2 kapal tanker lainnya berkapasitas 25.000 Kl). Urutan kapal tanker yang digunakan untuk mengirim solar diatur dengan mempertimbangkan prioritas kapal tanker agar penggunaan kapal tanker bisa maksimal sesuai dengan jumlah dan volume permintaan dari *customer*.



Pada penelitian ini dibuat dua model. Pada model pertama kapasitas dari 5 kapal tanker diasumsikan sama yaitu sebesar 22.000 KI karena sesuai data yang diambil dari penelitian sebelumnya yang dilakukan oleh Sierliawati (2014) yakni dari 26 permintaan tidak ada yang volumenya melebihi 22.000 KI. Pada model ini diprioritaskan menggunakan kapal tanker yang telah kembali dari pengiriman sebelumnya sehingga jumlah kapal tanker yang beroperasi dapat diminimumkan. Sedangkan pada model kedua banyaknya permintaan diasumsikan sama dengan pada model pertama namun 11 diantaranya dengan volume 22.000 KI-25.000 KI sehingga perbedaan kapasitas kapal tanker dipertimbangkan. Pada model ini pengiriman untuk volume permintaan tidak melebihi 22.000 KI diprioritaskan menggunakan kapal tanker dengan kapasitas 22.000 KI.

Berdasarkan model Aljabar *Max Plus* yang telah dibuat pada model pertama dan kedua serta dengan menggunakan data-data yang telah dikumpulkan selanjutnya dirancang jadwal *loading* dan keberangkatan kapal tanker. Jadwal dibuat menggunakan program Scilab.

## 1.2 Perumusan Masalah

Permasalahan yang akan dibahas dalam penelitian ini dinyatakan dalam rumusan masalah sebagai berikut:

1. Bagaimana membuat model rantai pasok menggunakan *Petri Net* dan Aljabar *Max Plus* dengan memprioritaskan pengiriman menggunakan kapal tanker yang telah kembali dari pengiriman sebelumnya?
2. Bagaimana merancang penjadwalan waktu *loading* dan keberangkatan kapal tanker menggunakan program Scilab dengan memprioritaskan pengiriman menggunakan kapal tanker yang telah kembali dari pengiriman sebelumnya?
3. Bagaimana membuat model rantai pasok menggunakan *Petri Net* dan Aljabar *Max Plus* dengan memprioritaskan pengiriman untuk volume permintaan tidak melebihi 22.000 KI menggunakan kapal tanker berkapasitas 22.000 KI?
4. Bagaimana merancang penjadwalan waktu *loading* dan keberangkatan kapal tanker menggunakan program Scilab dengan memprioritaskan

pengiriman untuk volume permintaan tidak melebihi 22.000 KI menggunakan kapal tanker berkapasitas 22.000 KI?

### 1.3 Batasan Masalah

Untuk membatasi ruang lingkup penelitian, maka dibuat batasan masalah sebagai berikut:

1. Rantai pasok distribusi solar pada penelitian ini melibatkan 1 *supplier* yaitu TBBM Tuban dan 2 *customer* yaitu dari TBBM Tuban menuju TBBM Manggis di Bali dan TBBM Tanjung Wangi di Banyuwangi
2. Data waktu dan volume permintaan diambil dari data pada penelitian sebelumnya (Sierliawati, 2014) yaitu data permintaan dari TBBM Tanjung Wangi dan TBBM Manggis pada tanggal 2 Oktober 2013-9 Desember 2013
3. Kapasitas (daya tampung) pengangkutan dari 5 kapal tanker untuk model *Petri Net* pertama dianggap sama yaitu 22.000 KI. Sedangkan pada model kedua perbedaan kapasitas pengangkutan 5 kapal tanker pengirim diperhatikan yaitu 3 kapal tanker mempunyai kapasitas pengangkutan 22.000 KI dan 2 kapal tanker yang lain mempunyai kapasitas pengangkutan 25.000 KI
4. Tidak ada gangguan perjalanan kapal dalam pengiriman solar
5. Waktu istirahat karyawan yang bertugas mengirim adalah saat proses *loading* produk solar
6. Performa mesin kapal tanker diasumsikan selalu stabil dan selalu ada karyawan yang siap untuk bertugas mengirimkan solar dengan kapal tanker

### 1.4 Tujuan penelitian

Adapun tujuan yang ingin dicapai dari penelitian ini adalah sebagai berikut:

1. Memperoleh model *Petri Net* dan Aljabar *Max Plus* dari rantai pasok dengan memprioritaskan pengiriman menggunakan kapal tanker yang telah kembali dari pengiriman sebelumnya

2. Memperoleh jadwal *loading* dan keberangkatan kapal tanker dengan memprioritaskan pengiriman menggunakan kapal tanker yang telah kembali dari pengiriman sebelumnya
3. Memperoleh model model *Petri Net* dan Aljabar *Max Plus* dari rantai pasok dengan memprioritaskan pengiriman untuk volume permintaan tidak melebihi 22.000 KI menggunakan kapal tanker berkapasitas 22.000 KI
4. Memperoleh jadwal *loading* dan keberangkatan kapal tanker dengan dengan memprioritaskan pengiriman untuk volume permintaan tidak melebihi 22.000 KI menggunakan kapal tanker berkapasitas 22.000 KI

### 1.5 Manfaat Penelitian

Dengan adanya penelitian ini, diharapkan dapat memberikan manfaat sebagai berikut:

1. Dapat mengembangkan aplikasi *Petri Net* dan Aljabar *Max Plus* dalam bidang industri manufaktur terutama pada model rantai pasok dari 1 *supplier* menuju 2 *customer* dengan volume permintaan berbeda-beda.
2. Dapat membantu *supplier* dalam mengatur penjadwalan distribusi atau pengiriman pasokan solar agar dapat memenuhi permintaan *customer* dengan tepat waktu sehingga kelangkaan akibat telatnya pengiriman bisa dihindari
3. Dapat membantu *customer* dalam membuat rencana waktu permintaan agar solar yang di pesan datang sesuai dengan waktu yang diharapkan.
4. Dengan adanya jadwal diharapkan karyawan yang bertugas mengirim solar pada *customer* bisa mengatur waktu untuk pengiriman dengan kecepatan kapal tanker yang sesuai standar kecepatan berkendara sehingga mengurangi resiko kecelakaan.
5. Dengan memprioritaskan pengiriman solar menggunakan kapal tanker yang telah kembali dari pengiriman sebelumnya maka jumlah kapal tanker yang digunakan untuk mengirim solar bisa diminimumkan.



## **BAB 2**

### **KAJIAN PUSTAKA DAN DASAR TEORI**

Pada bab ini dikaji beberapa penelitian-penelitian sebelumnya yang terkait dengan aplikasi *Petri Net* dan Aljabar *Max Plus* pada penjadwalan. Kemudian dibahas beberapa hal yang berkaitan dengan model rantai pasok distribusi solar pada PT. Pertamina serta dasar teori tentang *Petri Net* dan Aljabar *Max Plus*.

#### **2.1 Penelitian-Penelitian sebelumnya**

Aplikasi *Petri Net* dan Aljabar *Max Plus* untuk penjadwalan telah banyak dibuat pada beberapa penelitian. Adzkieya (2008) membuat model *Petri Net* untuk menentukan penjadwalan nyala lampu lalu lintas di persimpangan sehingga dapat memberikan kepastian waktu tunggu bagi para pengguna jalan, mengurangi kemacetan dan tingkat kecelakaan serta dapat memperlancar arus kendaraan. Model dibuat dengan mencantumkan waktu tunggu di persimpangan dengan jumlah jalur sebarang dalam bentuk algoritma.

Sedangkan penelitian tentang aplikasi *Petri Net* dan Aljabar *Max Plus* pada penjadwalan dalam rantai pasok antara lain penelitian yang dilakukan oleh Elmahi dkk. (2003) yang menggunakan model Aljabar *Max Plus* untuk merancang penjadwalan rantai pasok dari satu *supplier* menuju satu *customer* tanpa memperhatikan waktu *loading* dan *unloading* produk. Selanjutnya, Sierliawati (2014) menggunakan *Petri Net* untuk membuat model rantai pasok distribusi solar dari 1 *supplier* menuju 2 *customer* menggunakan 5 kapal tanker dengan memperhatikan waktu *loading* dan *unloading* produk. Model *Petri Net* yang dibuat menggunakan 1 token yang menunjukkan kapal tanker tidak memperhatikan prioritas kapal tanker.

Oleh karena itu, selanjutnya dilakukan penelitian ini yang berjudul “Model Rantai Pasok Menggunakan *Petri Net* dan Aljabar *Max Plus* dengan Mempertimbangkan Prioritas Kapal Tanker”. Rantai pasok yang dibuat yaitu distribusi solar dari TBBM Tuban menuju TBBM Manggis di Bali dan Tanjung

Wangi di Banyuwangi dengan menggunakan 5 kapal tanker. Pengiriman solar diprioritaskan menggunakan kapal tanker yang telah kembali dari pengiriman sebelumnya sehingga seluruh permintaan dapat dipenuhi hanya dengan menggunakan 4 kapal tanker dan diperoleh hasil yang sama dengan menggunakan 5 kapal tanker. Pada model *Petri Net* yang pertama, kapasitas kapal tanker dianggap sama. Sedangkan pada model *Petri Net* berikutnya, perbedaan kapasitas kapal tanker dipertimbangkan.

## **2.2 Definisi Solar**

Solar atau diesel fuel merupakan salah satu produk yang dihasilkan oleh PT. Pertamina yang bahan minyak mentahnya berasal dari pengeboran minyak dalam negeri. Warnanya jernih dan terang (Pertamina, 2012).

Solar atau minyak diesel adalah bahan bakar yang digunakan untuk motor diesel dimana proses pembakaran terjadi bukan oleh penyalan busi tetapi terjadi karena tekanan kompresi yang tinggi di dalam silinder motor. Minyak solar ini biasanya digunakan juga sebagai bahan bakar untuk pembakaran langsung di dalam dapur-dapur kecil yang menghendaki hasil pembakaran yang bersih (Depdiknas, 2003)

## **2.3 Definisi Rantai pasok**

Rantai pasok atau *supply chain* merupakan suatu sistem tempat organisasi menyalurkan barang produksi dan jasanya kepada para pelanggannya. Rantai ini juga merupakan jejaring dari berbagai organisasi yang saling berhubungan dengan tujuan yang sama, yaitu sebaik mungkin menyelenggarakan pengadaan atau penyaluran barang tersebut (Irawan, 2008).

*Supply chain* juga dapat dikatakan sebagai *logistics Network*, dengan pemain utama adalah :

1. *suppliers*,
2. *manufacturer*,
3. *distribution*,
4. *retail outlets*, dan
5. *customers* (Irawan, 2008).



## 2.4 Definisi Terminal Bahan Bakar Minyak (TBBM)

Terminal Bahan Bakar Minyak (TBBM) adalah tempat penimbunan dan penyaluran BBM yang dimiliki atau dikuasai oleh PT. Pertamina (Persero) dan/atau Badan Usaha lainnya yang mendapat penugasan (Sierliawati, 2014)

## 2.5 Wilayah Kerja Pertamina *Supply and Distribution Region III*

Pertamina *Supply and Distribution Region III* memiliki beberapa Terminal Bahan Bakar Minyak (TBBM), 6 TBBM di area Jawa Timur dan 12 TBBM di area Bali dan Nusa Tenggara. Berikut ini merupakan peta wilayah kerja PT. Pertamina *Supply and Distribution Region III*.



**Gambar 2.1** Peta Wilayah Kerja PT. Pertamina *Supply and Distribution Region III* (Sumber: Pertamina, 2012)

Berikut ini adalah keterangan penomoran pada Gambar 2.1:

- 6 TBBM Area Jawa Timur:
    - 1. TBBM Surabaya Group
    - 2. TBBM Tuban
    - 3. TBBM Malang\*
    - 4. TBBM Madiun\*
    - 5. TBBM Camplong
    - 6. TBBM Tanjung Wangi
- \*inland

- 12 TBBM Area Bali & Nusa Tenggara

- |                   |                   |
|-------------------|-------------------|
| 7. TBBM Manggis   | 13. TBBM Ende     |
| 8. TBBM Sanggaran | 14. TBBM Reo      |
| 9. TBBM Ampenan   | 15. TBBM Kalabahi |
| 10. TBBM Badas    | 16. TBBM Maumere  |
| 11. TBBM Bima     | 17. TBBM Tenau    |
| 12. TBBM Waingapu | 18. TBBM Atapupu  |

Rantai pasok yang dimodelkan dalam penelitian ini adalah rantai pasok distribusi BBM jenis solar dari TBBM Tuban sebagai menuju 2 TBBM sebagai *customer* yang berada di wilayah kerja Pertamina *Supply and Distribution Region III* yaitu TBBM Manggis di Bali dan TBBM Tanjung Wangi di Banyuwangi terjadi pada distribusi solar dari TBBM Tuban menuju TBBM Manggis dan TBBM Tanjung Wangi.

## 2.6 Aljabar *Max Plus*

Berikut ini diberikan pembahasan tentang definisi dan beberapa konsep dasar Aljabar *Max Plus*.

**Definisi Aljabar *Max Plus*** (Subiono, 2014):

Diberikan  $\mathbb{R}_\varepsilon \stackrel{\text{def}}{=} \mathbb{R} \cup \{\varepsilon\}$  dengan  $\mathbb{R}$  adalah himpunan semua bilangan real dan  $\varepsilon \stackrel{\text{def}}{=} -\infty$ . Pada  $\mathbb{R}_\varepsilon$  didefinisikan operasi berikut:  $\forall x, y \in \mathbb{R}_\varepsilon$ ,

$$x \oplus y \stackrel{\text{def}}{=} \max\{x, y\} \text{ dan } x \otimes y \stackrel{\text{def}}{=} x + y$$

Contoh:

$$10 \oplus 7 = \max\{10, 7\} = 10$$

$$10 \otimes 7 = 10 + 7 = 17$$

$(\mathbb{R}_\varepsilon, \oplus, \otimes)$  merupakan semiring dengan elemen netral  $\varepsilon$  dan elemen satuan  $e = 0$ .

Selanjutnya penulisan semiring  $(\mathbb{R}_\varepsilon, \oplus, \otimes)$  dituliskan sebagai  $\mathbb{R}_{\max}$ .

**Definisi Vektor dan Matriks pada Aljabar *Max Plus*** (Subiono, 2014):

Himpunan matriks ukuran  $n \times n$  dalam Aljabar *Max Plus* dinotasikan oleh  $\mathbb{R}_{\max}^{n \times m}$ .

Untuk  $n \in \mathbb{N}$  dengan  $n \neq 0$  didefinisikan  $\underline{n} \stackrel{\text{def}}{=} \{1, 2, \dots, n\}$ . Elemen  $A \in \mathbb{R}_{\max}^{n \times m}$

baris ke- $i$  kolom ke- $j$  dinotasikan oleh  $a_{ij}$  atau  $[A]_{ij}$  untuk  $i \in \underline{n}$  dan  $j \in \underline{m}$ .

Dalam hal ini matriks  $A$  ditulis sebagai berikut:

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & a_{2m} \\ \vdots & \vdots & \dots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nm} \end{bmatrix}$$

Ada kalanya elemen  $a_{ij}$  juga dinotasikan sebagai berikut:

$$[A]_{ij}, i \in \underline{n} \text{ dan } j \in \underline{m}$$

Matriks identitas dalam Aljabar *Max Plus* dinotasikan dengan  $E$  yang didefinisikan sebagai berikut:

$$[E(n, n)]_{i,j} \stackrel{\text{def}}{=} \begin{cases} e, & \text{untuk } i = j \\ \varepsilon, & \text{untuk } i \neq j \end{cases}$$

Penjumlahan matriks  $A, B \in \mathbb{R}_{\max}^{n \times m}$  yang dinotasikan oleh  $A \oplus B$  didefinisikan oleh:

$$[A \oplus B]_{i,j} \stackrel{\text{def}}{=} a_{i,j} \oplus b_{i,j}$$

Untuk perkalian matriks  $A \in \mathbb{R}_{\max}^{n \times m}$  dengan skalar  $\alpha \in \mathbb{R}_{\max}$  yang dinotasikan oleh  $\alpha \otimes A$  didefinisikan oleh:

$$[\alpha \otimes A]_{i,j} \stackrel{\text{def}}{=} \alpha \otimes a_{i,j}$$

Sedangkan untuk perkalian matriks  $A, B \in \mathbb{R}_{\max}^{n \times m}$  yang dinotasikan oleh  $A \otimes B$  didefinisikan oleh:

$$[A \otimes B]_{i,j} \stackrel{\text{def}}{=} \bigotimes_{k=1}^p a_{i,k} \otimes b_{k,j} \\ \stackrel{\text{def}}{=} \max\{A_{i,k} + B_{k,j}\}$$

Contoh:

Diberikan matriks

$$A = \begin{bmatrix} -6 & e \\ 4 & 3 \end{bmatrix} \text{ dan } B = \begin{bmatrix} 5 & -6 \\ \varepsilon & 1 \end{bmatrix},$$

maka

$$A \oplus B = \begin{bmatrix} -6 & e \\ 4 & 3 \end{bmatrix} \oplus \begin{bmatrix} 5 & -6 \\ \varepsilon & 1 \end{bmatrix} = \begin{bmatrix} -6 \oplus 5 & e \oplus -6 \\ 4 \oplus \varepsilon & 3 \oplus 1 \end{bmatrix} = \begin{bmatrix} 5 & e \\ 4 & 3 \end{bmatrix}$$

$$A \otimes B = \begin{bmatrix} -6 & e \\ 4 & 3 \end{bmatrix} \otimes \begin{bmatrix} 5 & -6 \\ \varepsilon & 1 \end{bmatrix} = \begin{bmatrix} -6 \otimes 5 \oplus e \otimes \varepsilon & -6 \otimes -6 \oplus e \otimes 1 \\ 4 \otimes 5 \oplus 3 \otimes \varepsilon & 4 \otimes -6 \oplus 3 \otimes 1 \end{bmatrix} = \begin{bmatrix} -1 & 1 \\ 9 & 4 \end{bmatrix}$$

$$-2 \otimes B = -2 \otimes \begin{bmatrix} 5 & -6 \\ \varepsilon & 1 \end{bmatrix} = \begin{bmatrix} -2 \otimes 5 & -2 \otimes -6 \\ -2 \otimes \varepsilon & -2 \otimes 1 \end{bmatrix} = \begin{bmatrix} 3 & -8 \\ \varepsilon & -1 \end{bmatrix}$$

**Definisi 2.1** (Subiono, 2014):

Untuk suatu matriks persegi  $A \in \mathbb{R}_{max}^{m \times n}$ , matriks  $A^+$  didefinisikan sebagai

$$A^+ \stackrel{\text{def}}{=} \bigotimes_{i=1}^{\infty} A^{\otimes i}, \text{ dengan } i = 1, 2, 3, \dots, +\infty$$

**Definisi 2.10** (Subiono, 2014):

$$A_{\lambda}^* \stackrel{\text{def}}{=} E \oplus A_{\lambda}^+ = \bigoplus_{i \geq 0} A_{\lambda}^{\oplus i}$$

dengan:

$$A_{\lambda}^+ \stackrel{\text{def}}{=} \bigotimes_{i=1}^{\infty} A_{\lambda}^{\otimes i} = A_{\lambda} \otimes A_{\lambda}^{\otimes 2} \otimes \dots \otimes A_{\lambda}^{\otimes n} \text{ dan}$$

$$A_{\lambda} \stackrel{\text{def}}{=} \lambda^{\otimes -1} \otimes A$$

**Teorema 2.3.1** (Subiono, 2014)

Misalkan  $A \in \mathbb{R}_{max}^{m \times n}$  adalah suatu matriks yang setiap kolomnya memuat setidaknya satu elemen tidak sama dengan  $\varepsilon$  dan  $b \in \mathbb{R}_{max}^m$ , maka

$$[x^*(A, b)]_j = \min\{b_i - a_{i,j} \mid i \in \underline{m} \text{ dan } a_{i,j} > \varepsilon\},$$

Bukti:

Perhatikan bahwa  $A \otimes x \leq b$  adalah ekuivalen dengan masing-masing berikut:

1. Untuk semua  $i$  dan  $j$ ,  $a_{i,j} + x_j \leq b_i$
2. Untuk semua  $i$  dan  $j$ ,  $x_j \leq b_i - a_{i,j}$  atau  $a_{i,j} = \varepsilon$
3. Untuk semua  $j$ ,  $x_j \leq \min\{b_i - a_{i,j} \mid i \in \underline{m} \text{ dan } a_{i,j} > \varepsilon\}$

Hal ini jelas bahwa  $x$  adalah suatu penyelesaian dari  $A \otimes x \leq b$  bila dan hanya bila untuk semua  $j$ ,  $x_j \leq \min\{b_i - a_{i,j} \mid i \in \underline{m} \text{ dan } a_{i,j} > \varepsilon\}$ . Oleh karena itu  $[x^*(A, b)]_j = \min\{b_i - a_{i,j} \mid i \in \underline{m} \text{ dan } a_{i,j} > \varepsilon\}$  adalah penyelesaian maksimum dari  $A \otimes x \leq b$ .

**Lemma 2.3.1** (Subiono, 2014)

Bila suatu penyelesaian dari  $A \otimes x \leq b$  ada, maka sub penyelesaian terbesar adalah penyelesaiannya.

Bukti:

Misalkan  $x'$  adalah suatu penyelesaian maksimum dari  $A \otimes x = b$ , maka  $x$  memenuhi pertaksamaan  $A \otimes x \leq b$ . Jadi haruslah  $x'$  adalah sub-penyelesaian terbesar. Sebagaimana diketahui sub-penyelesaian dari  $x^*(A, b)$  adalah maksimum penyelesaian dari  $A \otimes x \leq b$ . Karena penyelesaian dari  $A \otimes x = b$  ada, maka  $x^*(A, b)$  adalah penyelesaiannya. Hal ini menunjukkan bahwa sub-penyelesaian terbesar adalah suatu penyelesaian.

## 2.7 Petri Net

*Petri Net* dikembangkan pertama kali oleh C.A. Petri pada awal 1960-an. *Petri Net* merupakan salah satu alat untuk memodelkan sistem *event* diskrit selain menggunakan automata yang telah dikenal sebelumnya. Setiap automata dapat diubah menjadi *Petri Net*. Pada *Petri Net event* berkaitan dengan transisi. Agar suatu *event* dapat terjadi, beberapa keadaan harus dipenuhi terlebih dahulu. Informasi mengenai *event* dan keadaan ini masing-masing dinyatakan dengan transisi dan *Place*. *Place* dapat berfungsi sebagai input atau output suatu transisi. *Place* sebagai input menyatakan keadaan yang harus dipenuhi agar transisi dapat terjadi. Setelah transisi terjadi maka keadaan akan berubah. *Place* yang menyatakan keadaan tersebut adalah output dari transisi.

**Definisi 2.1:** (Adzkiya, 2008).

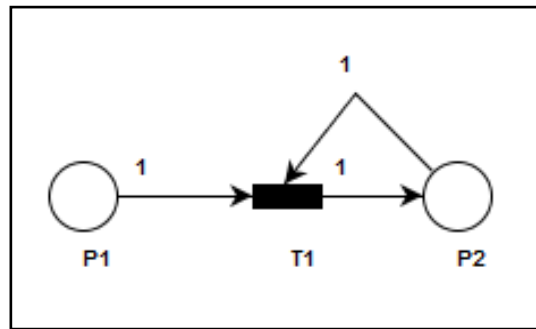
*Petri Net* adalah 4-tuple  $(P, T, A, w)$  dengan :

- $P$  : himpunan berhingga *place*,  $P = \{p_1, p_2, \dots, p_n\}$
- $T$  : himpunan berhingga transisi,  $T = \{t_1, t_2, \dots, t_n\}$ ,
- $A$  : himpunan *arc*,  $A \subseteq P = (P \times T) \cup (T \times P)$
- $w$  : fungsi bobot,  $w: A \rightarrow \{1, 2, 3, \dots\}$ .

Berdasarkan Definisi 2.1 maka himpunan *place* dan *transisi* tidak harus berupa himpunan berhingga melainkan bisa berupa himpunan tak berhingga

terhitung (*countable set*). *Petri Net* digambarkan sebagai *graph* berarah. Node dari *graph* berupa *place* yang diambil dari himpunan *place*  $P$  atau transisi yang diambil dari himpunan *transisi*  $T$ .

Grafik *Petri Net* terdiri dari dua macam node yaitu lingkaran dan garis. Lingkaran menyatakan *place* sedangkan garis menyatakan transisi. Arc yang menghubungkan *place*  $p_i$  ke transisi  $t_j$  berarti  $p_i \in I(t_j)$ . Jika bobot arc dari *place*  $p_i$  ke transisi  $t_j$  adalah  $k$  ditulis  $w(p_i, t_j) = k$  maka terdapat  $k$  arc dari *place*  $p_i$  ke transisi  $t_j$  atau sebuah arc dengan bobot  $k$ . Contoh *Petri Net* sederhana ditunjukkan pada Gambar 2.2 berikut.



**Gambar 2.2** Contoh *Petri Net* sederhana

Pada Gambar 2.2 terdapat 2 *place* yaitu  $p_1$  dan  $p_2$  ditulis  $P = \{p_1, p_2\}$  dan terdapat sebuah transisi yaitu  $t_1$  atau ditulis  $T = \{t_1\}$ . Arc dinyatakan dengan pasangan berurutan dengan elemen pertama menyatakan asal dan elemen kedua menyatakan tujuan misalnya arc dari *place*  $p_1$  ke transisi  $t_1$  maka ditulis  $(p_1, t_1)$ . Sedangkan bobot arc dari *place*  $p_1$  ke transisi  $t_1$  adalah 1 atau ditulis  $w(p_1, t_1) = 1$ .

**Definisi 2.2:** (Adzkiya, 2008).

*Petri Net* dikatakan pure jika tidak mempunyai *place* yang menjadi input sekaligus output untuk suatu transisi. Secara formal ditulis

$$\exists p_i \in P, t_j \in T \ni \{(p_i, t_j), (t_j, p_i),\}$$

Transisi pada *Petri Net* menyatakan *event* pada sistem *event* diskrit dan *place* merepresentasikan kondisi agar *event* dapat terjadi. Diperlukan mekanisme untuk



mengindikasikan apakah kondisi telah terpenuhi. Token adalah sesuatu yang diletakkan di *place* yang menyatakan terpenuhi tidaknya suatu kondisi. Secara grafik token digambarkan dengan dot dan diletakkan di dalam *place*. Jika jumlah token besar maka dituliskan dengan angka.

**Definisi 2.3:** (Adzkiya, 2008).

Penanda (marking)  $x$  pada *Petri Net* adalah fungsi  $x: P \rightarrow \{0,1,2, \dots\}$ , Penanda dinyatakan dengan vektor yang berisi bilangan bulat taknegatif yang menyatakan jumlah token yaitu  $x = [x(p_1), x(p_2), \dots, x(p_n)]^T$ . Jumlah elemen  $x$  sama dengan banyak *place* di *Petri Net*. Elemen ke- $i$  pada vektor ini merupakan jumlah token pada *place*  $p_i$ ,  $x(p_i) \in \{0,1,2, \dots\}$ .

**Definisi 2.4:** (Adzkiya, 2008).

*Petri Net* bertanda (marked) adalah 5-tuple  $(P, T, A, w, x_0)$  dimana  $(P, T, A, w)$  adalah *Petri Net* dan  $x_0$  adalah penanda awal. Selanjutnya *Petri Net* bertanda cukup disebut *Petri Net*. Seperti pemodelan sistem pada umumnya, maka harus didefinisikan keadaan(state) pada *Petri Net*. Keadaan pada *Petri Net* adalah penanda *Petri Net*.

**Definisi 2.5:** (Adzkiya, 2008).

Keadaan (state) *Petri Net* bertanda adalah  $x = [x(p_1), x(p_2), \dots, x(p_n)]^T$

Perhatikan bahwa jumlah token pada *place* adalah sebarang bilangan bulat taknegatif, tidak harus terbatas(bounded). Secara umum jumlah penanda yang mungkin adalah takhingga. Ruang keadaan(state space)  $X$  pada *Petri Net* bertanda dengan  $n$  *place* didefinisikan oleh semua vektor berdimensi  $n$  dengan elemen bilangan bulat tak negatif, sehingga  $X = \{0,1,2, \dots\}^n$ . Untuk selanjutnya digunakan istilah keadaan dan penanda bergantian.

Jika semua keadaan yang diperlukan sudah terpenuhi maka transisi dapat terjadi. Dalam hal ini keadaan merupakan *place* input dari transisi. Bobot arc dari *place* input ke transisi menunjukkan jumlah token minimum di *place* agar transisi *enabled*. Jika semua *place* input mempunyai token lebih dari atau sama dengan jumlah token minimum yang dibutuhkan maka transisi *enabled*.

**Dinamika *Petri Net*** (Adzkiya, 2008).

Jika *Petri Net* digunakan untuk memodelkan sistem dinamik *event* diskrit, seharusnya *Petri Net* dilengkapi dengan mekanisme yang mirip dengan transisi keadaan (*state transition*) pada automata. Mekanisme ini berupa menjalankan token melewati jaringan (*Net*) ketika transisi menjadi *enabled*. dan proses ini mengubah keadaan *Petri Net*.

Hanya transisi *enabled* yang dapat difire. Transisi difire saat *event* yang dinyatakan oleh transisi terjadi. Berikut ini adalah proses yang terjadi pada pemfirean transisi. Semua token di *place* input dikurangi/diambil sebanyak bobot arc yang menghubungkannya.

## **BAB 3**

### **METODE PENELITIAN**

Pada bab ini dibahas tentang tahapan penelitian yang dilakukan serta metode yang digunakan untuk mencapai tujuan penelitian. Adapun tahapan penelitian tersebut yaitu sebagai berikut:

1. Mempelajari teori Aljabar *Max Plus* dan *Petri Net*

Penelitian diawali dengan mempelajari teori-teori Aljabar *Max Plus* dan *Petri Net* serta mengkaji penelitian-penelitian sebelumnya yang berkaitan dengan penelitian yang akan dilakukan.

2. Pengumpulan data

Pada tahap ini, dilakukan pengumpulan beberapa data yang diperlukan untuk penelitian yaitu:

- a. Data dari PT. Pertamina yaitu data pembagian wilayah kerja Pertamina daerah *Supply and Region III* serta data kapasitas maksimal pengangkutan dari 5 kapal tanker yang beroperasi dari TBBM Tuban menuju TBBM Manggis di Bali dan TBBM Tanjung Wangi di Banyuwangi.
- b. Data yang diambil dari penelitian sebelumnya yaitu penelitian yang dilakukan oleh Sierliawati (2014) yang berupa data sekunder yaitu :
  - Data waktu permintaan
  - Data volume permintaan
  - Data waktu *loading* (dalam jam)
  - Data waktu *unloading* (dalam jam)
  - Data lama waktu berangkat pengiriman
  - Data lama waktu kembali dari pengiriman

3. Penyusunan model *Petri Net*.

Pada tahap ini dibuat model rantai pasok menggunakan *Petri Net*. Model *Petri Net* dibuat menggunakan program PIPE (*Platform Independent Petri Net Editor*) yaitu program yang digunakan untuk membuat *Petri Net* secara visual yang gratis. Pada model pertama kapasitas kapal tanker diasumsikan sama yakni 22.000 KI karena sesuai dengan data yang diambil yang volume

permintaannya tidak pernah melebihi 22.000 KI. Model pertama ini dibuat dengan memprioritaskan pengiriman solar menggunakan kapal tanker yang telah kembali dari pengiriman sebelumnya. Sedangkan model *Petri Net* yang kedua dibuat sebagai pengembangan dari model pertama yaitu untuk memodelkan rantai pasok dengan adanya kemungkinan permintaan yang volumenya sampai 25.000 KI sehingga perbedaan kapasitas kapal tanker dipertimbangkan. Pada model kedua dibuat pengaturan prioritas yaitu untuk volume kurang dari atau sama dengan 22.000 KI pengirimannya diprioritaskan menggunakan kapal tanker dengan kapasitas 22.000 KI.

#### 4. Penyusunan model Aljabar *Max Plus*

Pada tahap ini, dibuat model Aljabar *Max Plus* berdasarkan model *Petri Net* yang telah dibuat untuk model pertama dan model kedua.

#### 5. Penyusunan jadwal keberangkatan kapal tanker untuk model pertama

Pada tahap ini, data-data yang terkumpul yaitu data tanggal permintaan, lama proses *loading* dan *unloading*, serta lama berangkat dan kembali dari TBBM *customer* terlebih dahulu disimpan dalam file excel. Selanjutnya dibuat fungsi untuk mengkonversi data yaitu dari bentuk tanggal ke integer dan sebaliknya, serta dari tanggal ke bentuk string. Kemudian berdasarkan model Aljabar *Max Plus* yang telah dibuat selanjutnya dibuat fungsi untuk membuat jadwal *loading*, keberangkatan dan urutan dari 5 kapal tanker dengan memanggil data dari file excel yang telah tersimpan excel dan menggunakan fungsi untuk konversi data yang telah dibuat. Fungsi-fungsi tersebut dibuat dan disimulasikan pada program Scilab versi 5.4.1.

#### 6. Analisis hasil penjadwalan dari model pertama

Pada tahap ini dilakukan analisis terhadap jadwal yang telah diperoleh. Dengan mempertimbangkan waktu *loading* yang diperoleh dan waktu sampainya kapal tanker setelah melakukan pengiriman ke TBBM *customer*, maka penggunaan kapal tanker dapat diatur kembali. Sehingga selanjutnya dirancang jadwal baru dengan memprioritaskan pengiriman menggunakan kapal yang telah kembali dari pengiriman sebelumnya. Pengaturan urutan kapal tanker setelah diprioritaskan juga menggunakan program Scilab.

7. Penyusunan jadwal keberangkatan kapal tanker untuk model kedua

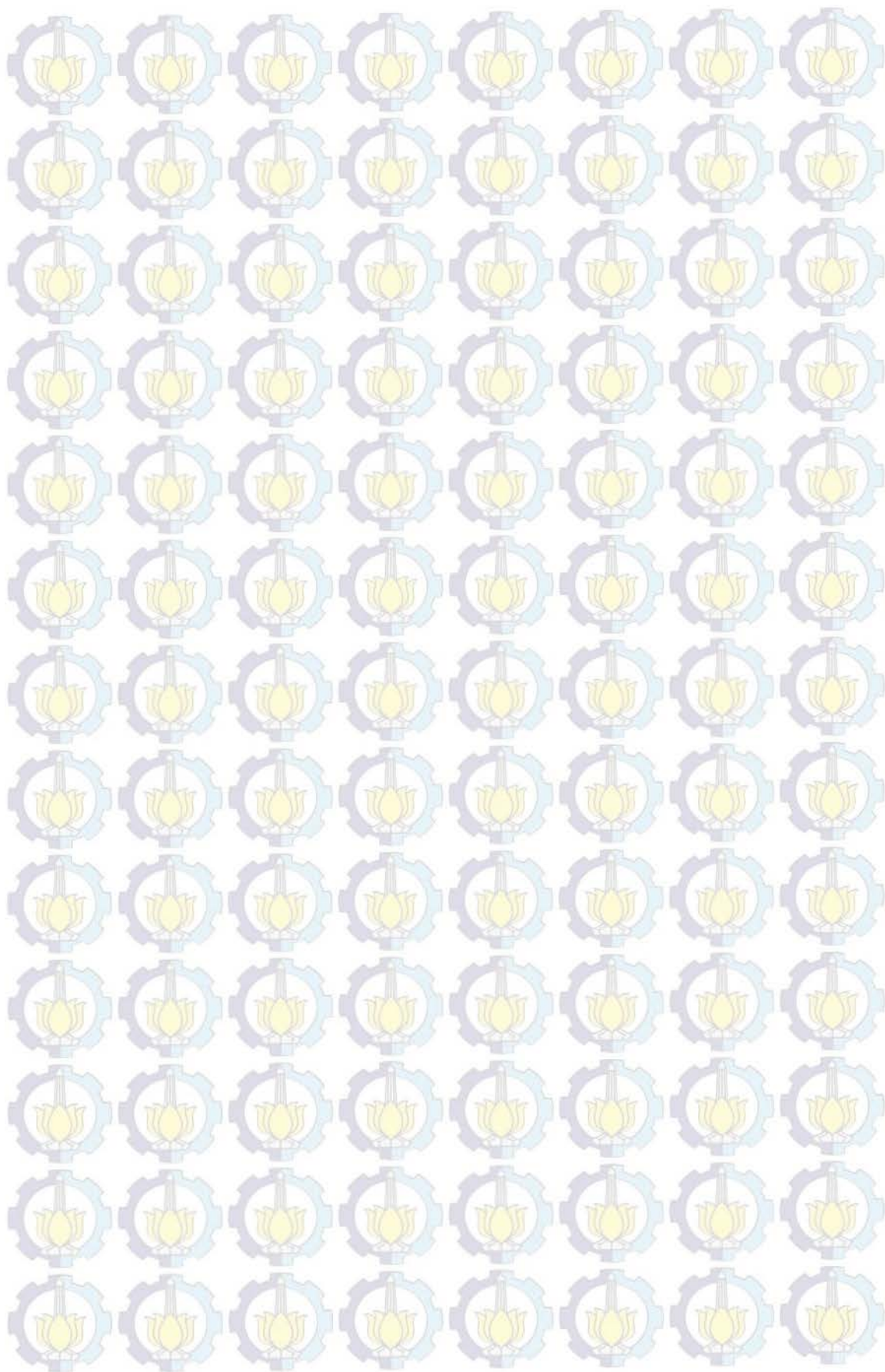
Untuk model kedua data-data yang berupa jumlah dan tanggal permintaan, lama *loading* dan *unloading* serta lama berangkat dan kembali sama dengan data pada model pertama. Sedangkan data volume permintaan diasumsikan ada 15 permintaan dengan volume kurang dari atau sama dengan 22.000 Kl dan 11 permintaan dengan volume 22.000 Kl-25.000 Kl. Seperti langkah-langkah pada model pertama data-data tersebut terlebih dahulu disimpan dalam file excel. Kemudian berdasarkan model Aljabar *Max Plus* yang telah dibuat selanjutnya dibuat fungsi untuk membuat jadwal *loading*, keberangkatan dan urutan dari 5 kapal tanker dengan memanggil data dari file excel yang telah tersimpan menggunakan fungsi-fungsi konversi seperti pada model pertama. Fungsi-fungsi selanjutnya disimulasikan program Scilab versi 5.4.1.

8. Diseminasi

Tahapan ini meliputi penulisan paper dan publikasi melalui kegiatan Seminar Nasional.

9. Penyusunan hasil penelitian

Pada tahap ini, dilakukan penulisan laporan hasil penelitian yang dilakukan mulai dari tahap studi literatur, pengumpulan data, penyusunan model *Petri Net* dan Aljabar *Max Plus* dari model distribusi dalam rantai pasok serta penjadwalan *loading* dan keberangkatan kapal tanker.





## BAB 4

### HASIL DAN PEMBAHASAN

Pada bab ini dibahas hasil pemodelan rantai pasok distribusi solar menggunakan *Petri Net* dan Aljabar *Max Plus* dengan mempertimbangkan prioritas kapal tanker serta penjadwalan waktu *loading* dan keberangkatan kapal tanker. Pada awal tahap dibahas pemodelan rantai pasok menggunakan *Petri Net* dengan kapasitas kapal tanker diasumsikan sama kemudian penjabaran ke model Aljabar *Max Plus*. Selanjutnya dibahas pembuatan jadwal *loading* dan keberangkatan dari 5 kapal tanker menggunakan program Scilab. Pada tahap berikutnya dibahas hasil analisis terhadap jadwal yang telah diperoleh untuk memprioritaskan pengiriman menggunakan kapal tanker yang telah kembali dari pengiriman sebelumnya. Pada tahap berikutnya ditunjukkan hasil pemodelan rantai pasok distribusi solar menggunakan *Petri Net* dengan mempertimbangkan perbedaan kapasitas kapal tanker. Model kedua ini dibuat untuk merancang kemungkinan adanya volume permintaan sebesar 22.000 Kl-25.000 Kl dan diprioritaskan pengiriman untuk volume tidak melebihi 22.000 Kl menggunakan kapal tanker berkapasitas 22.000 Kl. Selanjutnya model *Petri Net* tersebut dijabarkan dalam ke model Aljabar *Max Plus*. Pada tahap berikutnya dibahas pembuatan jadwal *loading* dan keberangkatan dari 5 kapal tanker menggunakan program Scilab.

#### **4.1. Model Rantai Pasok menggunakan *Petri Net* dengan Kapasitas Kapal Tanker dianggap sama**

Rantai pasok dalam penelitian ini adalah distribusi solar dari TBBM Tuban sebagai *supplier* menuju 2 *customer* yaitu TBBM Manggis di Bali dan Tanjung Wangi di Banyuwangi. Pengiriman Solar dilakukan melalui jalur laut dengan menggunakan kapal tanker milik PT. Pertamina. Adapun kapal tanker yang bisa digunakan untuk pengiriman pasokan solar di wilayah kerja PT. Pertamina tersebut sebanyak 5 dengan 3 kapal tanker berkapasitas 22.000 Kl dan 2 lainnya

25.000 Kl. Kelima kapal tanker tersebut adalah Fastron ex Tuban, Maiden, Pematang ex Tuban, Ae Pioner ex Tuban dan MT Pematang (Sierliawati, 2014).

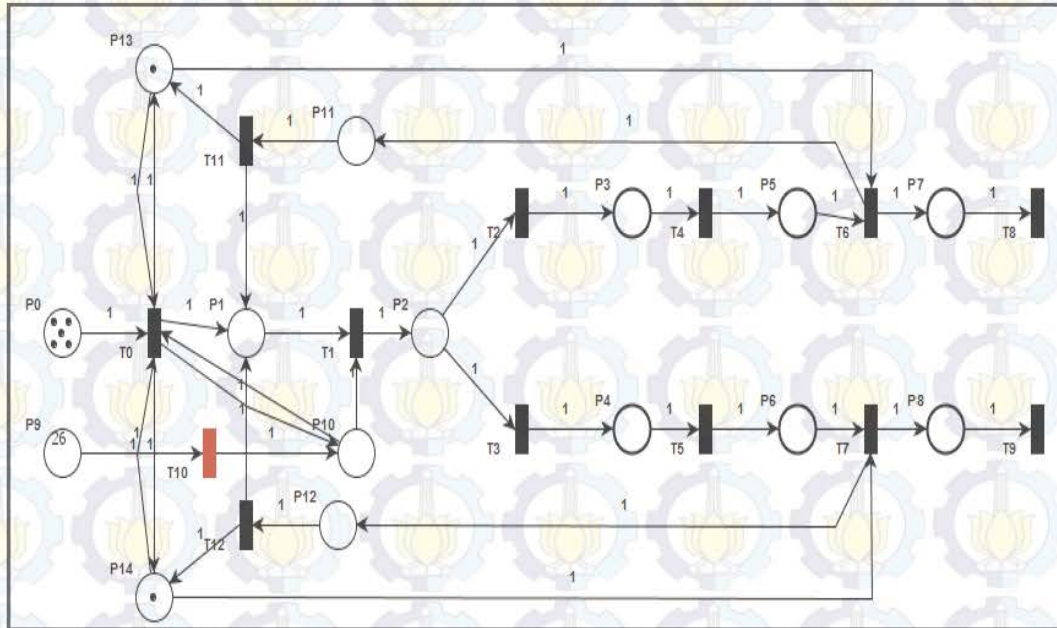
Pada model pertama ini kapasitas dari 5 kapal tanker dianggap sama yakni masing-masing sebesar 22.000 Kl. Hal ini disebabkan karena berdasarkan data yang diperoleh menunjukkan bahwa volume permintaan dari *customer* kurang dari atau sama dengan 22.000 Kl sehingga seluruh kapal tanker dapat digunakan untuk melakukan pengiriman solar. Pada model ini dibuat pengaturan prioritas penggunaan kapal tanker yaitu diprioritaskan melakukan pengiriman solar menggunakan kapal tanker yang telah kembali dari pengiriman sebelumnya sehingga diharapkan jumlah kapal tanker yang beroperasi dapat diminimumkan.

Alur pengiriman solar diawali dengan penetapan waktu dan volume permintaan dari TBBM Manggis dan Tanjung Wangi. Waktu permintaan dalam satuan menit, jam, tanggal, bulan dan tahun. Sedangkan volume permintaan dalam satuan Kilo liter (Kl). Selanjutnya TBBM Tuban mempersiapkan solar yang akan dikirim. Proses diawali dengan persiapan kapal tanker. Kemudian dilanjutkan dengan proses *loading*. Proses *loading* adalah pekerjaan pengangkutan BBM solar dari gudang ke kapal tanker (Sierliawati, 2014). Kemudian dilanjutkan dengan keberangkatan kapal tanker menuju TBBM *customer* yaitu TBBM Manggis di Bali atau Tanjung Wangi di Banyuwangi. Setelah sampai di TBBM *customer* selanjutnya dilakukan proses *unloading*. Proses *unloading* adalah pekerjaan yang dilakukan pada saat kapal tanker sampai pada *customer* yaitu pekerjaan pengangkutan BBM solar dari kapal tanker ke tangki (gudang) penyimpanan *customer* (Sierliawati, 2014). Selanjutnya solar diterima oleh TBBM *customer* dan kapal tanker kembali ke TBBM Tuban untuk melakukan pengiriman selanjutnya.

Rantai pasok distribusi solar ini dimodelkan menggunakan *Petri Net* dengan 14 *place* dan 12 transisi. Terdapat 5 token pada *place*  $P_0$  yang menunjukkan jumlah kapal tanker yang bisa digunakan untuk pengiriman solar yaitu sebanyak 5 kapal tanker. Pada *place*  $P_9$  terdapat 26 token yang menunjukkan solar siap dikirim untuk memenuhi 26 permintaan dari *customer*. Token pada *place*  $P_9$  ditulis dengan angka karena jumlahnya besar yaitu 26. Sedangkan pada *place*  $P_{13}$  dan  $P_{14}$  masing-masing memuat 1 token yang menunjukkan kapal



tanker baru siap digunakan karena tidak ada kapal tanker yang kembali dari TBBM *customer*. Jika tidak ada token di *place*  $P_{13}$  menunjukkan bahwa ada kapal tanker yang kembali dari TBBM Manggis dan jika tidak ada token di *place*  $P_{14}$  menunjukkan bahwa ada kapal tanker yang kembali dari Tanjung wangi. Keadaan awal model *Petri Net* dapat dilihat pada Gambar 4.1



**Gambar 4.1.** Keadaan awal Model *Petri Net* dengan Kapasitas Kapal Tanker dianggap sama

Keterangan:

*Place:*

$P_0$  = Kapal tanker

$P_1$  = Kapal tanker siap

$P_2$  = Proses *loading*

$P_3$  = Perjalanan ke TBBM Manggis

$P_4$  = Perjalanan ke TBBM Tanjung Wangi

$P_5$  = Proses *unloading* di TBBM Manggis

$P_6$  = Proses *unloading* di TBBM Tanjung Wangi

$P_7$  = Produk diterima di TBBM Manggis

$P_8$  = Produk diterima di TBBM Tanjung Wangi

$P_9$  = Produk solar (26 permintaan)

$P_{10}$  = Proses masuknya solar

$P_{11}$  = Perjalanan kembali dari TBBM Manggis

$P_{12}$  = Perjalanan kembali dari TBBM Tanjung Wangi

$P_{13}$  = Kapal tanker baru siap digunakan karena tidak ada yang kembali dari TBBM Manggis (ditunjukkan dengan adanya 1 token)

$P_{14}$  = Kapal tanker baru siap digunakan karena tidak ada yang kembali dari TBBM Tanjung Wangi (ditunjukkan dengan adanya 1 token)

Transisi:

$T_0$  = Kapal tanker mulai disiapkan

$T_1$  = Mulai proses *loading*

$T_2$  = Berangkat ke TBBM Manggis

$T_3$  = Berangkat ke TBBM Tanjung Wangi

$T_4$  = Mulai *unloading* di TBBM Manggis

$T_5$  = Mulai *unloading* di TBBM Tanjung Wangi

$T_6$  = Selesai *unloading* di TBBM Manggis

$T_7$  = Selesai *unloading* di TBBM Tanjung Wangi

$T_8$  = Produk di TBBM Manggis siap dipasarkan

$T_9$  = Produk di TBBM Tanjung Wangi siap dipasarkan

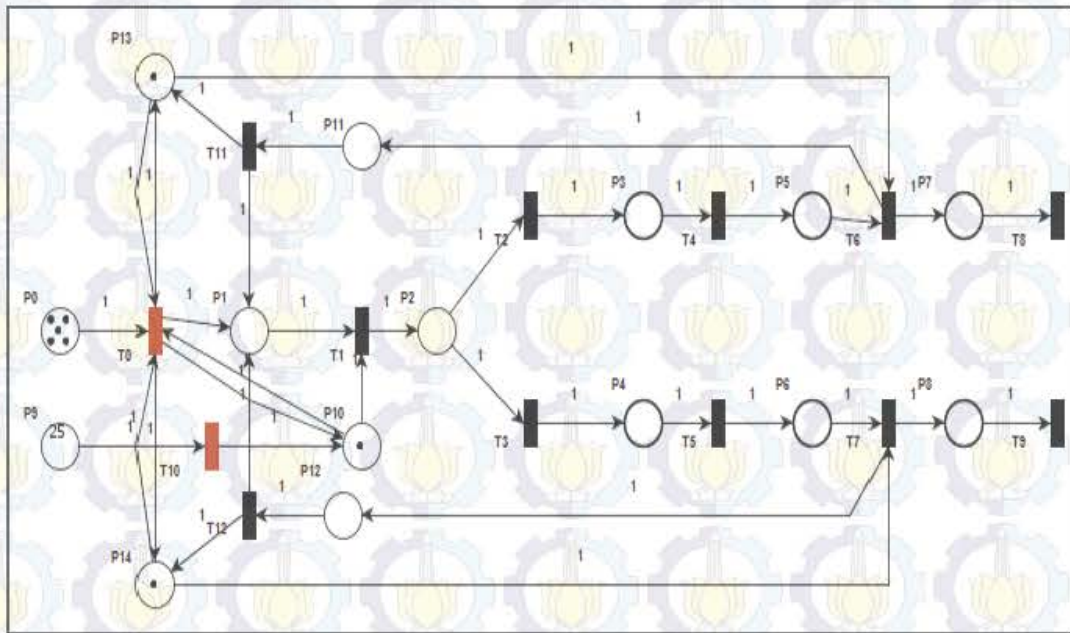
$T_{10}$  = Solar masuk

$T_{11}$  = Kapal tanker dari TBBM Manggis masuk TBBM Tuban

$T_{12}$  = Kapal tanker dari TBBM Tanjung Wangi masuk TBBM Tuban

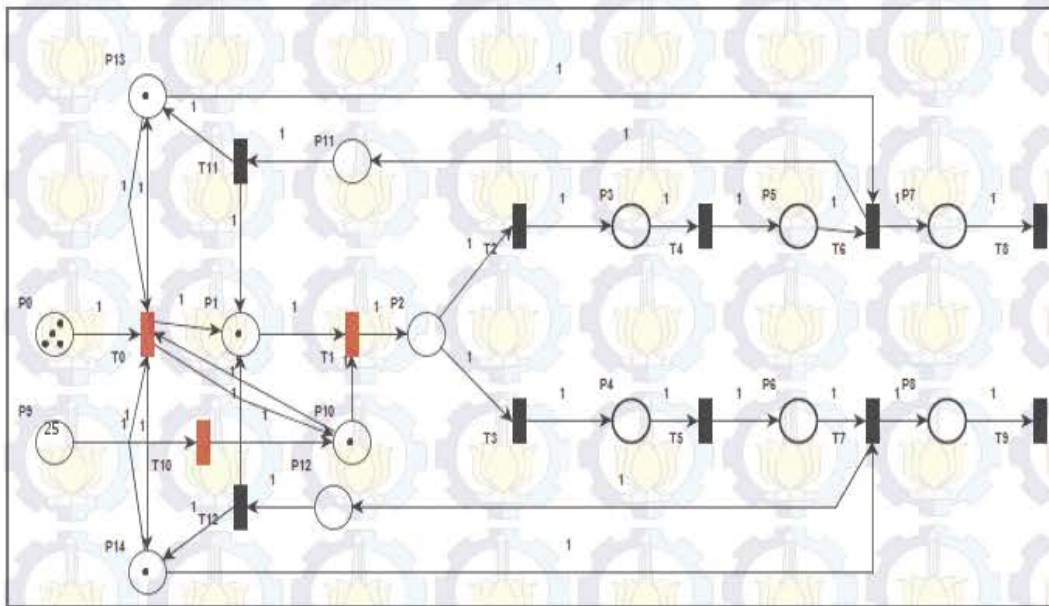
Gambar 4.1 menunjukkan keadaan awal *Petri Net* sebelum *difire*. Transisi  $T_{10}$  berwarna merah menunjukkan bahwa *enabled*. Transisi  $T_{10}$  *enabled* karena *place*  $P_9$  yang menjadi input dari transisi  $T_{10}$  mempunyai token lebih dari atau sama dengan jumlah token minimum yang dibutuhkan oleh transisi  $T_{10}$ . Hal ini menunjukkan solar yang akan dikirim untuk memenuhi 26 permintaan telah siap. Selanjutnya keadaan ketika transisi  $T_{10}$  telah *difire* sebanyak 1 kali dapat dilihat pada Gambar 4.2.





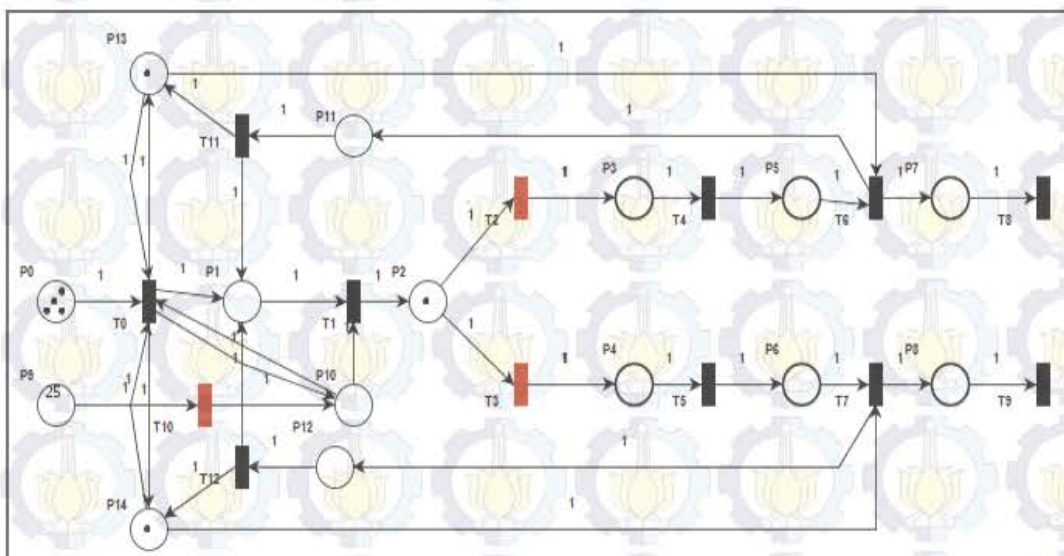
**Gambar 4.2.** Keadaan setelah Transisi  $T_{10}$  Difire sebanyak 1 Kali

Gambar 4.2 menunjukkan keadaan setelah transisi  $T_{10}$  difire sebanyak 1 kali yang menyebabkan transisi  $T_0$  menjadi *enabled* dan 1 token berpindah ke *place*  $P_{10}$ . Hal ini menunjukkan bahwa setelah ada solar yang siap untuk mulai proses *loading* maka kapal tanker juga bisa dipersiapkan. Sebelumnya pada Gambar 4.1 terlihat bahwa ketika *place*  $P_{10}$  tidak mempunyai token maka transisi  $T_0$  tidak *enabled*. Hal ini menunjukkan bahwa tidak ada solar untuk memenuhi permintaan *customer* maka kapal tanker juga tidak bisa dipersiapkan. Selanjutnya keadaan setelah transisi  $T_0$  difire dapat dilihat pada Gambar 4.3.



**Gambar 4.3.** Keadaan setelah Transisi  $T_0$  Difire sebanyak 1 Kali

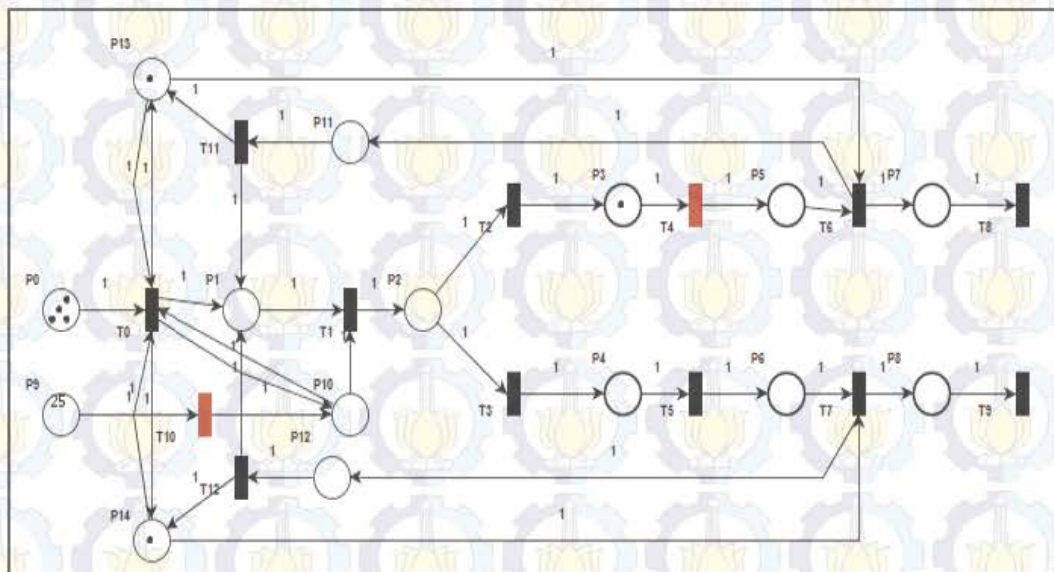
Pada Gambar 4.3 menunjukkan keadaan bahwa transisi  $T_1$  menjadi *enabled*. Hal ini menunjukkan bahwa pasokan solar untuk 1 permintaan dan sebuah kapal tanker telah dipersiapkan sehingga proses *loading* siap dimulai. Setelah transisi  $T_1$  difire, maka keadaan berikutnya adalah transisi  $T_2$  dan  $T_3$  menjadi *enabled* seperti yang terlihat pada Gambar 4.4.



**Gambar 4.4.** Keadaan setelah Transisi  $T_1$  Difire menyebabkan Transisi  $T_2$  dan  $T_3$  menjadi *enabled*

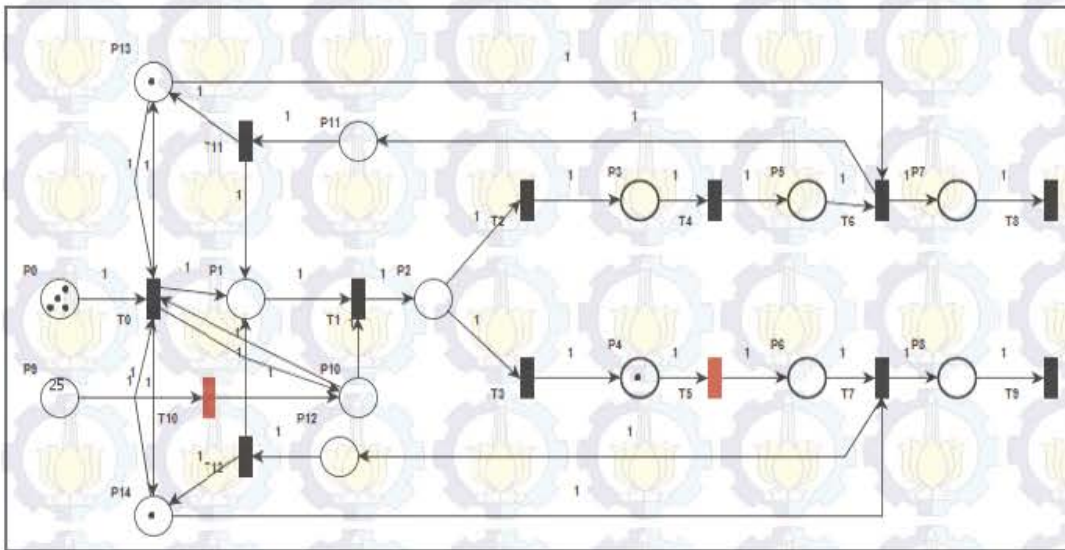


Gambar 4.4 menunjukkan bahwa proses *loading* telah selesai dan pasokan solar siap dikirim ke TBBM Manggis atau ke TBBM Tanjung wangi. Sedangkan transisi  $T_{10}$  menjadi tidak *enabled* lagi. Hal ini menunjukkan bahwa kapal tanker yang lain belum dipersiapkan untuk melakukan pengiriman berikutnya. Sehingga solar untuk memenuhi permintaan berikutnya juga belum siap untuk mulai *loading*. Selanjutnya keadaan ketika transisi  $T_2$  *difire* ditunjukkan pada Gambar 4.5.



**Gambar 4.5.** Keadaan setelah Transisi  $T_2$  *Difire* menunjukkan keberangkatan Kapal Tanker menuju TBBM Manggis

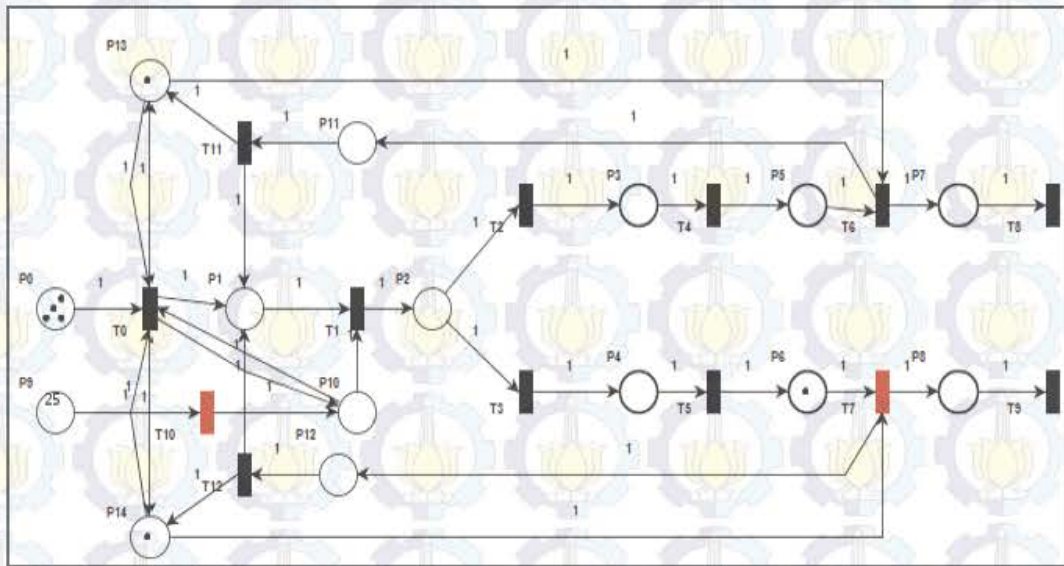
Gambar 4.5 menunjukkan sebuah kapal tanker berangkat menuju TBBM Manggis. Ketika transisi  $T_2$  *difire* maka transisi  $T_4$  menjadi *enabled*. Hal ini menunjukkan bahwa proses *unloading* siap dilakukan setelah kapal sampai di TBBM Manggis. Sedangkan transisi  $T_3$  menjadi tidak *enabled*. Hal ini menunjukkan bahwa sebuah kapal akan berangkat hanya ke salah satu TBBM dan tidak mungkin ke keduanya sekaligus. Sebuah kapal yang telah berangkat ke TBBM Manggis tidak mungkin bisa sekaligus berangkat ke TBBM Tanjung Wangi. Sedangkan keadaan ketika dilanjutkan transisi  $T_3$  *difire* ditunjukkan pada Gambar 4.6.



**Gambar 4.6.** Keadaan setelah Transisi  $T_3$  Difire menunjukkan Keberangkatan Kapal Tanker menuju TBBM Tanjung Wangi

Gambar 4.6 menunjukkan bahwa kapal tanker berangkat menuju TBBM Tanjung Wangi sehingga transisi  $T_2$  menjadi tidak *enabled*. Ketika transisi  $T_3$  difire maka transisi  $T_5$  menjadi *enabled*. Hal ini menunjukkan bahwa proses *unloading* siap dilakukan setelah kapal sampai di TBBM Tanjung Wangi. Sedangkan transisi  $T_2$  menjadi tidak *enabled*. Hal ini menunjukkan bahwa sebuah kapal tanker yang telah berangkat ke TBBM Tanjung Wangi juga tidak mungkin bisa sekaligus berangkat ke TBBM Manggis. Selanjutnya keadaan ketika transisi  $T_5$  difire ditunjukkan pada Gambar 4.7.

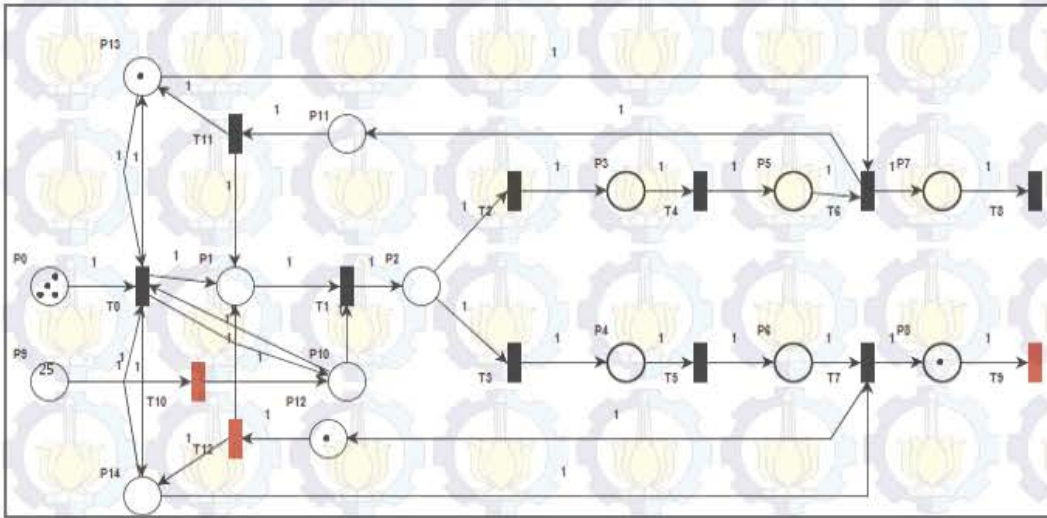




**Gambar 4.7.** Keadaan setelah Transisi  $T_5$  Difire

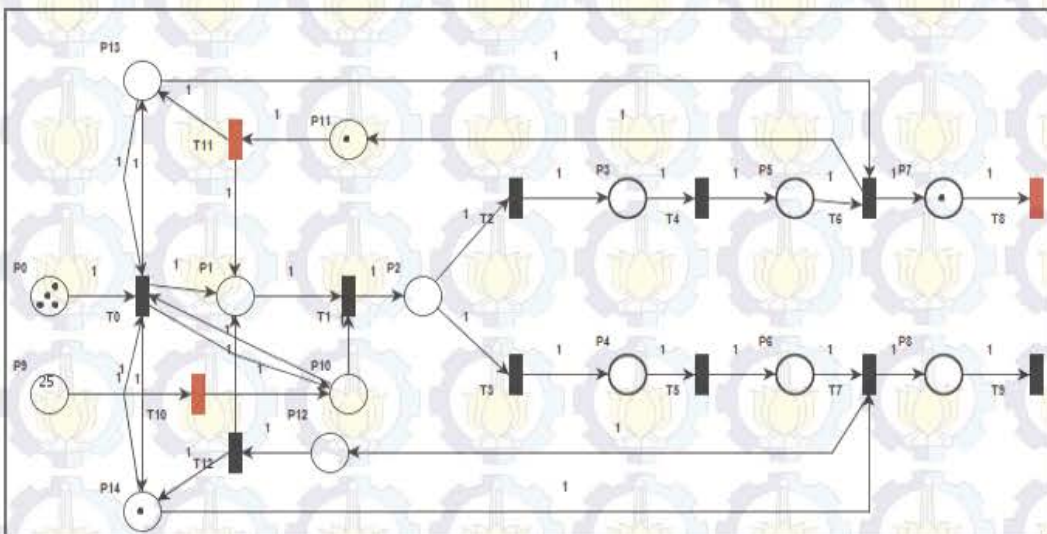
Gambar 4.7 menunjukkan keadaan setelah transisi  $T_5$  difire. Hal ini menunjukkan bahwa sedang terjadi proses *unloading* di TBBM Tanjung Wangi. Setelah transisi  $T_5$  difire maka transisi  $T_7$  menjadi *enabled*.

Adanya masing-masing 1 token pada *place*  $P_{13}$  dan *place*  $P_{14}$  menunjukkan bahwa tidak ada kapal tanker yang kembali dari TBBM Manggis dan TBBM Tanjung Wangi, sehingga bisa menggunakan kapal tanker yang berada di *place*  $P_0$ . Akan tetapi, ketika transisi  $T_6$  telah difire maka tidak ada token pada *place*  $P_{13}$  atau ketika transisi  $T_7$  telah difire maka tidak ada token pada *place*  $P_{14}$  sehingga *place*  $P_0$  tidak *enabled*. Hal ini menunjukkan ketika ada kapal tanker yang kembali dari TBBM Manggis atau dari TBBM Tanjung Wangi maka pengiriman berikutnya diprioritaskan menggunakan kapal tanker yang telah kembali dari TBBM Manggis atau TBBM Tanjung Wangi tersebut sehingga tidak perlu menggunakan kapal tanker yang berada di *place*  $P_0$ . Keadaan setelah transisi  $T_7$  difire ditunjukkan pada Gambar 4.8.



**Gambar 4.8.** Keadaan setelah Transisi  $T_7$  Difire menyebabkan  $T_0$  tidak *Enabled*

Gambar 4.8 menunjukkan keadaan setelah proses *unloading* di TBBM Tanjung Wangi telah selesai dan solar telah diterima oleh TBBM Tanjung Wangi. Transisi  $T_9$  menjadi *enabled*. Hal ini menunjukkan bahwa produk solar sedang dipersiapkan untuk dipasarkan. Sedangkan transisi  $T_0$  tidak *enabled* karena adanya kapal tanker yang telah kembali dari TBBM Tanjung Wangi sehingga diprioritaskan pengiriman selanjutnya menggunakan kapal tanker yang telah kembali tersebut, yang ditunjukkan dengan keadaan dimana transisi  $T_{12}$  menjadi *enabled*. Keadaan setelah transisi  $T_6$  difire ditunjukkan pada Gambar 4.9.



**Gambar 4.9.** Keadaan setelah Transisi  $T_6$  Difire menyebabkan  $T_0$  tidak *Enabled*



Gambar 4.9 menunjukkan bahwa transisi  $T_0$  tidak *enabled* karena adanya kapal tanker yang telah kembali dari TBBM Manggis sehingga diprioritaskan pengiriman selanjutnya menggunakan kapal tanker yang telah kembali tersebut, yang ditunjukkan dengan keadaan dimana transisi  $T_{11}$  menjadi *enabled*.

#### 4.2. Model Aljabar *Max Plus* dari Model *Petri Net* dengan Kapasitas Kapal Tanker dianggap sama

Model *Petri Net* yang telah dibuat selanjutnya dijabarkan dalam model Aljabar *Max Plus*. Berdasarkan model *Petri Net* yang telah dibentuk menunjukkan bahwa waktu *loading* untuk permintaan ke-1 sampai ke-5 langsung bisa segera dilakukan setelah produk solar siap karena kapal tanker yang tersedia sebanyak 5. Sedangkan waktu *loading* untuk permintaan ke-6 dan seterusnya harus menunggu kapal tanker kembali dari pengiriman sehingga waktu *loading* juga dipengaruhi oleh sampainya kapal tanker di TBBM *customer* dan lamanya perjalanan dari TBBM *customer* pada permintaan sebelumnya. Waktu sampainya produk di TBBM *customer* merupakan waktu selesainya proses *unloading* sehingga dipengaruhi oleh waktu *loading*, lama proses *loading*, lama keberangkatan menuju TBBM *customer* dan lama proses *unloading*. Dan waktu diterimanya produk solar di TBBM *customer* sesuai dengan waktu selesainya proses *unloading*. Berdasarkan keterangan yang diperoleh dari model *petri Net* tersebut maka diperoleh model Aljabar *Max Plus* sebagai berikut:

$$t_m(k) = \begin{cases} t_n(k-n) \otimes p_d(k-n) \oplus u(k), & \text{untuk } k > n \\ u(k), & \text{untuk } 1 \leq k \leq n \end{cases} \quad (4.1)$$

$$t_n(k) = t_m(k) \otimes p_a(k) \otimes p_b(k) \otimes p_c(k) \quad (4.2)$$

$$y(k) = t_n(k) \quad (4.3)$$

Keterangan:

$n$  = banyaknya kapal tanker

$t_m(k)$  = waktu *loading* permintaan ke- $k$  ( $T_1(k)$ )

$t_n(k)$  = waktu sampai permintaan ke- $k$  (meliputi  $T_6(k)$  dan  $T_7(k)$ )

$y(k)$  = waktu produk diterima saat ke- $k$  (meliputi  $T_6(k)$  dan  $T_7(k)$ )

$p_a(k)$  = lama *loading* permintaan ke- $k$  ( $P_2(k)$ )

$p_b(k)$  = lama berangkat permintaan ke- $k$  (meliputi  $P_3(k)$  dan  $P_4(k)$ )

$p_c(k)$  = lama *unloading* permintaan ke- $k$  (meliputi  $P_5(k)$  dan  $P_6(k)$ )

$p_d(k)$  = lama kembali saat ke- $k$  (meliputi  $P_{11}(k)$  dan  $P_{12}(k)$ )

$u(k)$  = waktu masuknya solar saat ke- $k$  ( $T_{10}(k)$ )

Selanjutnya dengan memisalkan:

$$t(k) = \begin{bmatrix} t_m(k) \\ t_n(k) \end{bmatrix}, A_0(k) = \begin{bmatrix} \varepsilon & \varepsilon \\ p_a(k) \otimes p_b(k) \otimes p_c(k) & \varepsilon \end{bmatrix},$$

$$A_1(k) = \begin{bmatrix} \varepsilon & p_d(k-n) \\ \varepsilon & \varepsilon \end{bmatrix}, B_0 = \begin{bmatrix} e \\ \varepsilon \end{bmatrix}, \text{ dan } C = [\varepsilon \quad e], \text{ dengan } e = 0$$

Maka Persamaan (4.1) dan (4.3) dapat ditulis menjadi Persamaan (4.5) sebagai berikut:

$$t(k) = A_0(k) \otimes t(k) \oplus A_1(k) \otimes t(k-n) \oplus B_0 \otimes u(k) \quad (4.5)$$

Substitusi  $t(k)$  pada ruas kiri Persamaan (4.5) ke ruas kanannya sehingga diperoleh persamaan sebagai berikut:

$$t(k) = A_0^2(k) \otimes t(k) \oplus (E \oplus A_0(k)) \otimes A_1(k) \otimes t(k-n) \oplus (E \oplus A_0(k)) \otimes B_0 \otimes u(k) \quad (4.6)$$

dengan  $E = \begin{bmatrix} 0 & \varepsilon \\ \varepsilon & 0 \end{bmatrix}$  yang merupakan matriks identitas dengan ukuran  $2 \times 2$ .

Kemudian substitusikan kembali  $t(k)$  pada ruas kiri Persamaan (4.6) ke ruas kanannya sehingga diperoleh persamaan sebagai berikut:

$$t(k) = A_0^3(k) \otimes t(k) \oplus (E \oplus A_0(k) \oplus A_0^2(k)) \otimes A_1(k) \otimes t(k-n) \oplus (E \oplus A_0(k) \oplus A_0^2(k)) \otimes B_0 \otimes u(k)$$

Ulangi lagi substitusi sebanyak  $\beta$  sehingga diperoleh persamaan sebagai berikut:

$$t(k) = A_0^\beta(k) \otimes t(k) \oplus (E \oplus A_0(k) \oplus \dots \oplus A_0^{\beta-1}(k)) \otimes A_1(k) \otimes t(k-n) \oplus (E \oplus A_0(k) \oplus \dots \oplus A_0^{\beta-1}(k)) \otimes B_0 \otimes u(k) \quad (4.7)$$

Lalu untuk  $\beta$  mendekati tak hingga maka Persamaan (4.7) menjadi persamaan baru sebagai berikut:

$$t(k) = (E \oplus A_0(k) \oplus A_0^2(k) \oplus \dots) \otimes A_1(k) \otimes t(k-n) \oplus (E \oplus A_0(k) \oplus A_0^2(k) \oplus \dots) \otimes B_0 \otimes u(k) \quad (4.8)$$

Berdasarkan Definisi 2.10 maka Persamaan (4.8) dapat ditulis menjadi Persamaan (4.9) sebagai berikut:

$$t(k) = A_0^*(k) \otimes A_1(k) \otimes t(k-n) \oplus A_0^*(k) \otimes B_0 \otimes u(k) \quad (4.9)$$



Selanjutnya Persamaan (4.9) dapat ditulis menjadi Persamaan (4.10) sebagai berikut:

$$t(k) = A(k) \otimes t(k-n) \oplus B(k) \otimes u(k) \quad (4.10)$$

dengan:

$$A(k) = A_0^*(k) \otimes A_1(k) \text{ dan}$$

$$B(k) = A_0^*(k) \otimes B_0$$

Sedangkan Persamaan (4.4) dapat ditulis menjadi Persamaan (4.11) sebagai berikut:

$$y(k) = C \otimes t(k) \quad (4.11)$$

Selanjutnya dengan mensubstitusi  $t(k)$  yang terdapat pada Persamaan (4.10) ke dalam Persamaan (4.11) maka diperoleh persamaan sebagai berikut:

$$y(k) = C \otimes A(k) \otimes t(k-n) \oplus C \otimes B(k) \otimes u(k) \quad (4.12)$$

Kemudian substitusikan kembali  $t(k)$  pada Persamaan (4.10) ke Persamaan (4.12) sehingga diperoleh persamaan sebagai berikut:

$$y(k) = C \otimes A(k) \otimes A(k-n) \otimes t(k-2n) \oplus C \otimes A(k) \otimes B(k-n) \otimes u(k-n) \oplus C \otimes B(k) \otimes u(k)$$

Ulangi lagi substitusi sebanyak  $\alpha$  sehingga diperoleh persamaan sebagai berikut:

$$y(k) = [C \otimes (A(k) \otimes A(k-n) \otimes \dots \otimes A(k-(\alpha-1)n)) \otimes t(k-\alpha n)] \oplus [C \otimes B(k) \otimes u(k)] \oplus [C \otimes A(k) \otimes B(k-n) \otimes u(k-n)] \oplus [C \otimes A(k) \otimes A(k-n) \otimes B(k-2n) \otimes u(k-2n)] \oplus \dots \oplus [C \otimes (A(k) \otimes A(k-n) \otimes \dots \otimes A(k-(\alpha-2)n)) \otimes B(k-(\alpha-1)n) \otimes u(k-(\alpha-1)n)] \quad (4.13)$$

Kemudian untuk  $k > n$  maka Persamaan (4.2) dan (4.3) dapat ditulis menjadi Persamaan (4.14) sebagai berikut:

$$t(k) = A_0(k) \otimes t(k) \oplus B_0 \otimes u(k) \quad (4.14)$$

Substitusi  $t(k)$  pada ruas kiri Persamaan (4.14) ke ruas kanannya sehingga diperoleh persamaan sebagai berikut:

$$t(k) = A_0^2(k) \otimes t(k) \oplus A_0(k) \otimes B_0 \otimes u(k) \oplus B_0 \otimes u(k) \quad (4.15)$$

Kemudian substitusikan kembali  $t(k)$  pada ruas kiri Persamaan (4.15) ke ruas kanannya sehingga diperoleh persamaan sebagai berikut:

$$t(k) = A_0^3(k) \otimes t(k) \oplus A_0^2(k) \otimes B_0 \otimes u(k) \oplus A_0(k) \otimes B_0 \otimes u(k) \oplus B_0 \otimes u(k)$$



Ulangi lagi substitusi sebanyak  $\beta$  sehingga diperoleh persamaan sebagai berikut:

$$t(k) = A_0^\beta(k) \otimes t(k) \oplus (E \oplus A_0(k) \oplus \dots \oplus A_0^{\beta-1}(k)) \otimes B_0 \otimes u(k) \quad (4.16)$$

Lalu untuk  $\beta$  mendekati tak hingga maka Persamaan (4.16) menjadi persamaan baru sebagai berikut:

$$t(k) = (E \oplus A_0(k) \oplus A_0^2(k) \oplus \dots) \otimes B_0 \otimes u(k) \quad (4.17)$$

Berdasarkan Definisi 2.10 maka Persamaan (4.17) dapat ditulis menjadi Persamaan (4.18) sebagai berikut:

$$t(k) = A_0^*(k) \otimes B_0 \otimes u(k) \quad (4.18)$$

Selanjutnya Persamaan (4.18) dapat ditulis menjadi Persamaan (4.19) sebagai berikut:

$$t(k) = B(k) \otimes u(k) \quad (4.19)$$

dengan:

$$B(k) = A_0^*(k) \otimes B_0$$

Berdasarkan Persamaan (4.19) maka diperoleh persamaan sebagai berikut:

$$t(k - \alpha n) = B(k - \alpha n) \otimes u(k - \alpha n) \quad (4.20)$$

Selanjutnya substitusi Persamaan (4.20) ke Persamaan (4.13) sehingga diperoleh persamaan baru sebagai berikut:

$$\begin{aligned} y(k) = & [C \otimes B(k) \otimes u(k)] \oplus [C \otimes A(k) \otimes B(k - n) \otimes u(k - n)] \oplus [C \otimes A(k) \otimes \\ & A(k - n) \otimes B(k - 2n) \otimes u(k - 2n)] \oplus \dots \oplus [C \otimes (A(k) \otimes A(k - n) \otimes \dots \\ & \otimes A(k - (\alpha - 2)n)) \otimes B(k - (\alpha - 1)n) \otimes u(k - (\alpha - 1)n)] \oplus [C \otimes (A(k) \\ & \otimes A(k - n) \otimes \dots \otimes A(k - (\alpha - 1)n)) \otimes B(k - \alpha n) \otimes u(k - \alpha n)] \end{aligned} \quad (4.21)$$

Selanjutnya berdasarkan Persamaan 4.21 maka dapat dibentuk persamaan sebagai berikut:

$$Y = H \otimes U \quad (4.22)$$

dengan elemen-elemen pada matriks  $Y$  diperoleh dari data waktu permintaan yang telah ditetapkan oleh TBBM *customer*, elemen-elemen pada matriks  $H$  diperoleh berdasarkan hasil perhitungan data-data sekunder yang telah terkumpul dan elemen-elemen matrik  $U$  merupakan waktu mulai *loading* yang akan dihitung.

Untuk banyaknya permintaan dinyatakan dengan  $n_d$ , banyaknya kapal tanker dinyatakan dengan  $n$  dan  $x = l + \gamma n$  menyatakan urutan permintaan

dimulai dari 0 dengan  $\gamma$  merupakan hasil bagi  $x$  terhadap  $n$  dan  $l$  merupakan sisa pembagian  $x$  terhadap  $n$  atau  $x$  modulo  $n$ , maka diperoleh penjabaran bentuk matriks  $Y, H$  dan  $U$  sebagai berikut:

$$Y = \begin{bmatrix} y(l) \\ y(l+n) \\ y(l+2n) \\ \vdots \\ y(l+\gamma n) \end{bmatrix}, U = \begin{bmatrix} u(l) \\ u(l+n) \\ u(l+2n) \\ \vdots \\ u(l+\gamma n) \end{bmatrix}, \text{ dan}$$

$$H = \begin{bmatrix} CB(l) & \varepsilon & \dots & \varepsilon \\ CA(l+n)B(l) & CB(l+n) & \dots & \varepsilon \\ \vdots & \vdots & \ddots & \vdots \\ CA(l+n)A(l+2n) \dots A(l+\gamma n)B(l) & \dots & \dots & CB(l+\gamma n) \end{bmatrix}$$

Keterangan:

$$\begin{aligned} A(l+n) &= A_0^*(l+n) \otimes A_1(l+n) \\ &= (E \oplus A_0(l+n) \oplus A_0^2(l+n) \oplus \dots) \otimes A_1(l+n) \\ &= \left( \begin{bmatrix} e & \varepsilon \\ \varepsilon & e \end{bmatrix} \oplus \left[ p_a(l+n) \otimes p_b(l+n) \otimes p_c(l+n) \right] \oplus \begin{bmatrix} \varepsilon & \varepsilon \\ \varepsilon & \varepsilon \end{bmatrix} \oplus \dots \right) \otimes \begin{bmatrix} \varepsilon & p_d(l) \\ \varepsilon & \varepsilon \end{bmatrix} \\ &= \begin{bmatrix} \varepsilon & p_d(l) \\ \varepsilon & p_a(l+n) \otimes p_b(l+n) \otimes p_c(l+n) \end{bmatrix} \\ B(l) &= A_0^*(l) \otimes B_0 \\ &= \begin{bmatrix} e & \varepsilon \\ p_a(l) \otimes p_b(l) \otimes p_c(l) & e \end{bmatrix} \otimes \begin{bmatrix} e \\ \varepsilon \end{bmatrix} \\ &= \begin{bmatrix} e & \\ p_a(l) \otimes p_b(l) \otimes p_c(l) & \end{bmatrix} \end{aligned}$$

Pada penelitian ini jumlah permintaan 26 sehingga  $n_d = 26$ . Jumlah kapal tanker yang bisa digunakan untuk mengirim solar adalah 5 sehingga  $n = 5$ . Urutan permintaan adalah  $x = 0, 1, 2, \dots, 25$  dengan  $x = 0$  menunjukkan permintaan ke-1,  $x = 1$  menunjukkan permintaan ke-2, dan seterusnya sampai  $x = 25$  menunjukkan permintaan ke-26 sehingga  $k = x + 1$ . Sedangkan  $\gamma$  merupakan hasil bagi  $x$  terhadap 5 dan  $l = 0, 1, 2, 3, 4$  merupakan sisa pembagian  $x$  terhadap 5 atau  $x$  modulo 5.



Selanjutnya diperoleh matriks  $Y$  dalam urutan permintaan yang sebenarnya ( $k$ ) dan dalam  $x$  sebagai berikut:

$$Y(k) \rightarrow Y(x)$$

$$\begin{bmatrix} y(1) \\ y(2) \\ y(3) \\ y(4) \\ y(5) \\ y(6) \\ y(7) \\ y(8) \\ y(9) \\ y(10) \\ y(11) \\ y(12) \\ y(13) \\ y(14) \\ y(15) \\ y(16) \\ y(17) \\ y(18) \\ y(19) \\ y(20) \\ y(21) \\ y(23) \\ y(24) \\ y(25) \\ y(26) \end{bmatrix} \rightarrow \begin{bmatrix} y(0) \\ y(1) \\ y(2) \\ y(3) \\ y(4) \\ y(5) \\ y(6) \\ y(7) \\ y(8) \\ y(9) \\ y(10) \\ y(11) \\ y(12) \\ y(13) \\ y(14) \\ y(15) \\ y(16) \\ y(17) \\ y(18) \\ y(19) \\ y(20) \\ y(21) \\ y(23) \\ y(24) \\ y(25) \end{bmatrix} = \begin{bmatrix} y(0) \\ y(1) \\ y(2) \\ y(3) \\ y(4) \\ y(0 + 1.5) \\ y(1 + 1.5) \\ y(2 + 1.5) \\ y(3 + 1.5) \\ y(4 + 1.5) \\ y(0 + 1.5) \\ y(1 + 2.5) \\ y(2 + 2.5) \\ y(3 + 2.5) \\ y(4 + 2.5) \\ y(0 + 3.5) \\ y(1 + 3.5) \\ y(2 + 3.5) \\ y(3 + 3.5) \\ y(4 + 3.5) \\ y(0 + 4.5) \\ y(1 + 4.5) \\ y(2 + 4.5) \\ y(3 + 4.5) \\ y(4 + 4.5) \end{bmatrix}$$

Untuk  $x$  modulo 5 = 0 atau  $l = 0$  menunjukkan bahwa permintaan dipenuhi oleh kapal tanker ke-1. maka  $x = 5.n$  dengan  $n = 0,1,2,3,4,5$  sehingga  $k = x + 1 = 5.n + 1$  dan diperoleh:

$$Y1 = \begin{bmatrix} y(1) \\ y(6) \\ y(11) \\ y(16) \\ y(21) \\ y(26) \end{bmatrix} \text{ dan}$$

$$H1 = \begin{bmatrix} CB(1) & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ CA(6)B(1) & CB(6) & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ CA(11)A(6)B(1) & CA(11)B(6) & CB(11) & \varepsilon & \varepsilon & \varepsilon \\ CA(16)A(11)A(6)B(1) & CA(16)A(11)B(6) & CA(16)B(11) & CB(16) & \varepsilon & \varepsilon \\ CA(21)A(16)A(11)A(6)B(1) & CA(21)A(16)A(11)B(6) & CA(21)A(16)B(11) & CA(21)B(16) & CB(21) & \varepsilon \\ CA(26)A(21)A(16)A(11)A(6)B(1) & CA(26)A(21)A(16)A(11)B(6) & CA(26)A(21)A(16)B(11) & CA(26)A(21)B(16) & CA(26)B(21) & CB(26) \end{bmatrix}$$

Untuk  $x$  modulo 5 = 1 atau  $l = 1$  menunjukkan bahwa permintaan dipenuhi oleh kapal tanker ke-2. maka  $x = 5.n + 1$  dengan  $n = 0,1,2,3,4$  sehingga  $k = x + 1 = 5.n + 1 + 1 = 5.n + 2$  dan diperoleh:

$$Y2 = \begin{bmatrix} y(2) \\ y(7) \\ y(12) \\ y(17) \\ y(22) \end{bmatrix}, \text{ dan}$$

$$H2 = \begin{bmatrix} CB(2) & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ CA(7)B(2) & CB(7) & \varepsilon & \varepsilon & \varepsilon \\ CA(12)A(7)B(2) & CA(12)B(7) & CB(12) & \varepsilon & \varepsilon \\ CA(17)A(12)A(7)B(2) & CA(17)A(12)B(7) & CA(17)B(12) & CB(17) & \varepsilon \\ CA(22)A(17)A(12)A(7)B(2) & CA(22)A(17)A(12)B(7) & CA(22)A(17)B(12) & CA(22)B(17) & CB(22) \end{bmatrix}$$

Untuk  $x$  modulo 5 = 2 atau  $l = 2$  menunjukkan bahwa permintaan dipenuhi oleh kapal tanker ke-3. maka  $x = 5.n + 2$  dengan  $n = 0,1,2,3,4$  sehingga  $k = x + 1 = 5.n + 2 + 1 = 5.n + 3$  dan diperoleh:

$$Y3 = \begin{bmatrix} y(3) \\ y(8) \\ y(13) \\ y(18) \\ y(23) \end{bmatrix}, \text{ dan}$$

$$H3 = \begin{bmatrix} CB(3) & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ CA(8)B(3) & CB(8) & \varepsilon & \varepsilon & \varepsilon \\ CA(13)A(8)B(3) & CA(13)B(8) & CB(13) & \varepsilon & \varepsilon \\ CA(18)A(13)A(8)B(3) & CA(18)A(13)B(8) & CA(18)B(13) & CB(18) & \varepsilon \\ CA(23)A(18)A(13)A(8)B(3) & CA(23)A(18)A(13)B(8) & CA(23)A(18)B(13) & CA(23)B(18) & CB(23) \end{bmatrix}$$

Untuk  $x$  modulo 5 = 3 atau  $l = 3$  menunjukkan bahwa permintaan dipenuhi oleh kapal tanker ke-3. maka  $x = 5.n + 3$  dengan  $n = 0,1,2,3,4$  sehingga  $k = x + 1 = 5.n + 3 + 1 = 5.n + 4$  dan diperoleh:

$$Y4 = \begin{bmatrix} y(4) \\ y(9) \\ y(14) \\ y(19) \\ y(24) \end{bmatrix}, \text{ dan}$$



$$H4 = \begin{bmatrix} CB(4) & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ CA(9)B(4) & CB(9) & \varepsilon & \varepsilon & \varepsilon \\ CA(14)A(9)B(4) & CA(14)B(9) & CB(14) & \varepsilon & \varepsilon \\ CA(19)A(14)A(9)B(4) & CA(19)A(14)B(9) & CA(19)B(14) & CB(19) & \varepsilon \\ CA(24)A(19)A(14)A(9)B(4) & CA(24)A(19)A(14)B(9) & CA(24)A(19)B(14) & CA(24)B(19) & CB(24) \end{bmatrix}$$

Untuk  $x$  modulo 5 = 4 atau  $l = 4$  menunjukkan bahwa permintaan dipenuhi oleh kapal tanker ke-5. maka  $x = 5.n + 4$  dengan  $n = 0,1,2,3,4$  sehingga  $k = x + 1 = 5.n + 4 + 1 = 5.n + 5$  dan diperoleh:

$$Y5 = \begin{bmatrix} y(5) \\ y(10) \\ y(15) \\ y(20) \\ y(25) \end{bmatrix} \text{ dan}$$

$$H5 = \begin{bmatrix} CB(5) & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ CA(10)B(5) & CB(10) & \varepsilon & \varepsilon & \varepsilon \\ CA(15)A(10)B(5) & CA(15)B(10) & CB(15) & \varepsilon & \varepsilon \\ CA(20)A(15)A(10)B(5) & CA(20)A(15)B(10) & CA(20)B(15) & CB(20) & \varepsilon \\ CA(25)A(20)A(15)A(10)B(5) & CA(25)A(20)A(15)B(10) & CA(25)A(20)B(15) & CA(25)B(20) & CB(25) \end{bmatrix}$$

Selanjutnya nilai dari elemen-elemen matriks  $Y$  dan  $H$  untuk masing-masing  $l = 0,1,2,3,4$  atau  $Y1$  dan  $H1$ ,  $Y2$  dan  $H2$ ,  $Y3$  dan  $H3$ ,  $Y4$  dan  $H4$ , serta  $Y5$  dan  $H5$  dihitung pada program Scilab dengan menggunakan data-data sekunder yang telah dikumpulkan.

#### 4.3 Data-Data sekunder untuk Model pertama yang digunakan dalam Simulasi Program

Data sekunder yang berupa data permintaan dari TBBM Manggis dan TBBM Tanjung Wangi dapat dilihat pada Lampiran 1.A yang meliputi data volume permintaan dalam satuan Kilo liter (KI) dan waktu permintaan dalam bentuk menit, jam, tanggal, bulan dan tahun. Data tersebut telah diurutkan sesuai dengan tanggal permintaan. Terdapat 26 permintaan dari dua TBBM *customer*. Permintaan dari TBBM Manggis sebanyak 15 permintaan. Sedangkan permintaan dari TBBM Tanjung Wangi sebanyak 11 permintaan.

Adapun lamanya keberangkatan ke TBBM Manggis atau lama kembali adalah 22 jam. Sedangkan lamanya keberangkatan ke TBBM Tanjung Wangi atau

kembali adalah 16,2 jam. Lamanya waktu *loading* dan *unloading* untuk masing-masing permintaan dapat dihitung dari volume permintaan dibagi kecepatan rata-rata waktu angkut kapal tanker (yaitu 500 KI per jam).

Data-data sekunder tersebut terlebih dahulu disimpan dalam file excel. Hal ini untuk memudahkan ketika disimulasikan dalam program Scilab. Adapun data yang disimpan dalam file excel dapat dilihat pada Lampiran 1.B. Data menit, jam, tanggal, bulan dan tahun permintaan disimpan di *sheet* 1. Data waktu *loading* dan *unloading* adalah sama sehingga cukup disimpan satu data pada *sheet* 2. Sedangkan data lama berangkat atau kembali dari TBBM *customer* disimpan dalam *sheet* 3. Data-data tersebut dalam penelitian ini disimpan dalam file excel dengan nama data\_P. Data-data tersebut kemudian digunakan untuk membuat jadwal *loading* dan keberangkatan kapal tanker pada program Scilab.

#### **4.4 Penjadwalan Model pertama dengan Program Scilab**

Sebelum dibuat jadwal terlebih dahulu dibuat fungsi dalam program Scilab untuk mengkonversi data. Ada tiga fungsi yang dibuat untuk mengkonversi data yaitu fungsi untuk mengkonversi data dari bentuk jam dan tanggal permintaan ke bentuk integer, dari bentuk integer ke bentuk jam dan tanggal serta dari bentuk jam dan tanggal ke bentuk string. Fungsi untuk mengkonversi data jam dan tanggal ke bentuk integer dalam penelitian ini diberi nama fungsi *date2intetc*. Adapun konversinya terhitung sejak pukul 01.01 tanggal 1 Januari dan tahun sesuai inputan. Sedangkan fungsi untuk mengkonversi integer ke bentuk menit, jam, tanggal, bulan dan tahun dalam penelitian ini diberi nama fungsi *int2dateetc*. Dan fungsi untuk mengkonversi data tanggal ke bentuk string diberi nama fungsi *date2stretc*. Listing program untuk fungsi *date2intetc*, *int2dateetc* dan *date2stretc* dapat dilihat berturut-turut pada Lampiran 2.A, 2.B dan 3.A.

Selanjutnya dibuat fungsi untuk membuat jadwal waktu *loading* dan keberangkatan pengiriman solar menggunakan 5 kapal tanker. Fungsi dibuat dalam program Scilab dan diberi nama fungsi *supplychain1to2*. Adapun listing program untuk fungsi *supplychain1to2* dapat dilihat pada Lampiran 3.B.

Dalam fungsi tersebut diawali dengan pembentukan matriks  $Y$  dan  $H$  berdasarkan bentuk matriks yang telah diperoleh dari model Aljabar *Max Plus*.



Data-data yang sudah tersimpan dalam file excel dipanggil. Data lama *loading* atau *unloading* yang tersimpan dalam *sheet* 2 dan data berangkat atau kembali dari TBBM *customer* yang tersimpan dalam *sheet* 3 dalam satuan jam terlebih dahulu dikonversi ke bentuk menit dengan cara dikalikan dengan 60.

Berikut ini merupakan matriks  $Y$  dan  $H$  untuk  $l = 1,2,3,4,5$  berdasarkan hasil simulasi dalam program Scilab:

Y1=

395100.  
423900.  
441180.  
459900.  
475740.  
493020.

Y2=

400860.  
425340.  
446940.  
464220.

Y3=

412380.  
429660.  
449820.  
464220.

Y4=

416700.  
435420.  
451260.  
467100.

Y5=

423900.

435420.

452700.

475740.

H1 =

6600. - Inf - Inf - Inf - Inf

14520. 6600. - Inf - Inf - Inf

18252. 10332. 2412. - Inf - Inf

21396. 13476. 5556. 2172. - Inf

28968. 21048. 13128. 9744. 6600. - Inf

34368. 26448. 18528. 15144. 12000. 4080.

H2=

4200. - Inf - Inf - Inf

8892. 3372. - Inf - Inf

14064. 8544. 4200. - Inf

19716. 14196. 9852. 4332. - Inf

22848. 17328. 12984. 7464. 2160.

H3=

4200. - Inf - Inf - Inf

12120. 6600. - Inf - Inf

17532. 12012. 4092. - Inf

25104. 19584. 11664. 6600. - Inf

30036. 24516. 16596. 11532. 3612.



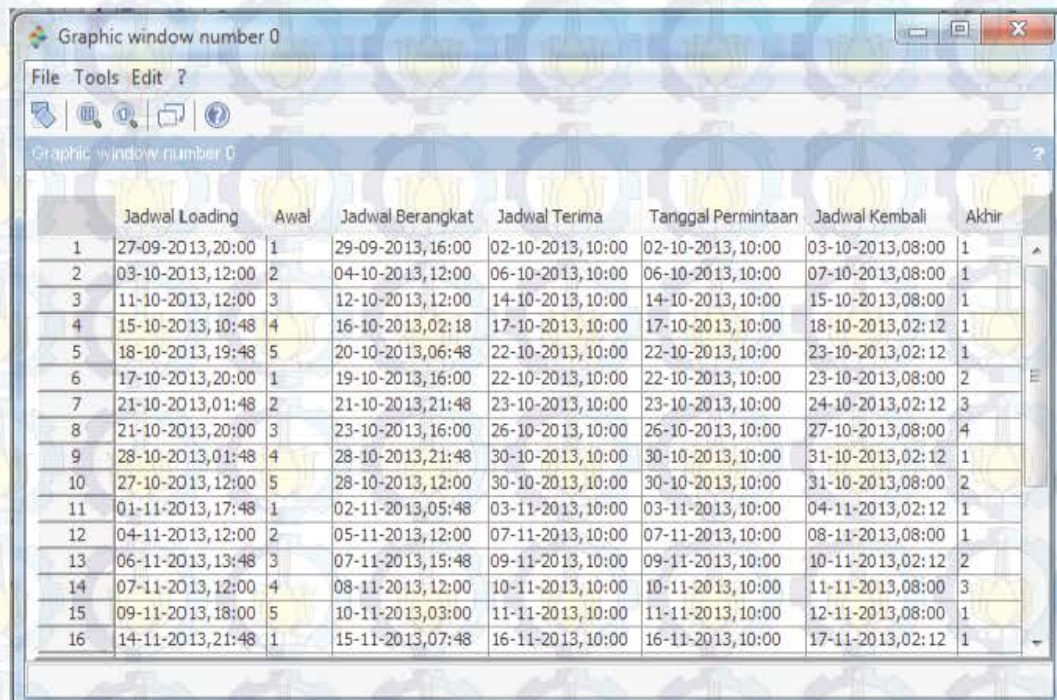
H4=

2832.	- Inf	- Inf	- Inf	- Inf
7176.	3372.	- Inf	- Inf	- Inf
12348.	8544.	4200.	- Inf	- Inf
15828.	12024.	7680.	2160.	- Inf
21348.	17544.	13200.	7680.	4200.

H5=

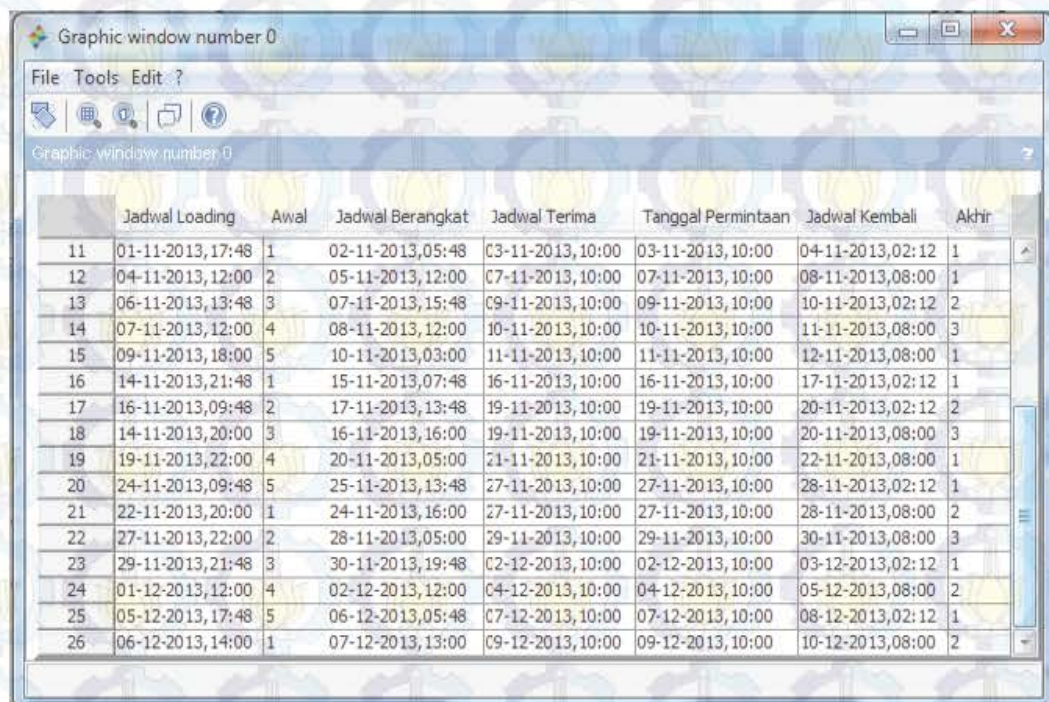
5172.	- Inf	- Inf	- Inf	- Inf
10344.	4200.	- Inf	- Inf	- Inf
14064.	7920.	2400.	- Inf	- Inf
19716.	13572.	8052.	4332.	- Inf
23100.	16956.	11436.	7716.	2412.

Setelah terbentuk matriks  $Y$  dan  $H$  selanjutnya disusun suatu persamaan  $Y = H \otimes U$  untuk masing-masing  $l = 0,1,2,3,4$  yaitu  $Y1 = H1 \otimes U1$ ,  $Y2 = H2 \otimes U2$ ,  $Y3 = H3 \otimes U3$ ,  $Y4 = H4 \otimes U4$  dan  $Y5 = H5 \otimes U5$  sehingga dapat dihitung nilai  $U$  atau waktu mulai *loading* yaitu  $U1, U2, U3, U4$  dan  $U5$  berdasarkan Teorema 2.3.1 dan Lemma 2.3.1 yang dalam program Scilab bisa menggunakan fungsi `maxpluslinsol`. Setelah mendapatkan nilai  $U$  selanjutnya dapat dihitung waktu mulai *loading*, keberangkatan kapal, sampainya kapal di TBBM *customer* dan di TBBM *supplier* setelah kembali dari pengiriman. Selain itu, dengan menggunakan program Scilab dapat juga dibuat urutan kapal tanker untuk mengirim solar sebelum dan setelah diprioritaskan menggunakan kapal tanker yang telah kembali dari pengiriman sebelumnya. Adapun hasil simulasi dari fungsi `supplychain1to2` pada program Scilab berupa penjadwalan yang ditunjukkan pada Gambar 4.8. dan Gambar 4.9.



	Jadwal Loading	Awal	Jadwal Berangkat	Jadwal Terima	Tanggal Permintaan	Jadwal Kembali	Akhir
1	27-09-2013, 20:00	1	29-09-2013, 16:00	02-10-2013, 10:00	02-10-2013, 10:00	03-10-2013, 08:00	1
2	03-10-2013, 12:00	2	04-10-2013, 12:00	06-10-2013, 10:00	06-10-2013, 10:00	07-10-2013, 08:00	1
3	11-10-2013, 12:00	3	12-10-2013, 12:00	14-10-2013, 10:00	14-10-2013, 10:00	15-10-2013, 08:00	1
4	15-10-2013, 10:48	4	16-10-2013, 02:18	17-10-2013, 10:00	17-10-2013, 10:00	18-10-2013, 02:12	1
5	18-10-2013, 19:48	5	20-10-2013, 06:48	22-10-2013, 10:00	22-10-2013, 10:00	23-10-2013, 02:12	1
6	17-10-2013, 20:00	1	19-10-2013, 16:00	22-10-2013, 10:00	22-10-2013, 10:00	23-10-2013, 08:00	2
7	21-10-2013, 01:48	2	21-10-2013, 21:48	23-10-2013, 10:00	23-10-2013, 10:00	24-10-2013, 02:12	3
8	21-10-2013, 20:00	3	23-10-2013, 16:00	26-10-2013, 10:00	26-10-2013, 10:00	27-10-2013, 08:00	4
9	28-10-2013, 01:48	4	28-10-2013, 21:48	30-10-2013, 10:00	30-10-2013, 10:00	31-10-2013, 02:12	1
10	27-10-2013, 12:00	5	28-10-2013, 12:00	30-10-2013, 10:00	30-10-2013, 10:00	31-10-2013, 08:00	2
11	01-11-2013, 17:48	1	02-11-2013, 05:48	03-11-2013, 10:00	03-11-2013, 10:00	04-11-2013, 02:12	1
12	04-11-2013, 12:00	2	05-11-2013, 12:00	07-11-2013, 10:00	07-11-2013, 10:00	08-11-2013, 08:00	1
13	06-11-2013, 13:48	3	07-11-2013, 15:48	09-11-2013, 10:00	09-11-2013, 10:00	10-11-2013, 02:12	2
14	07-11-2013, 12:00	4	08-11-2013, 12:00	10-11-2013, 10:00	10-11-2013, 10:00	11-11-2013, 08:00	3
15	09-11-2013, 18:00	5	10-11-2013, 03:00	11-11-2013, 10:00	11-11-2013, 10:00	12-11-2013, 08:00	1
16	14-11-2013, 21:48	1	15-11-2013, 07:48	16-11-2013, 10:00	16-11-2013, 10:00	17-11-2013, 02:12	1

**Gambar 4.10.** Hasil Simulasi Program berupa Penjadwalan untuk permintaan ke-1 sampai ke-16



	Jadwal Loading	Awal	Jadwal Berangkat	Jadwal Terima	Tanggal Permintaan	Jadwal Kembali	Akhir
11	01-11-2013, 17:48	1	02-11-2013, 05:48	03-11-2013, 10:00	03-11-2013, 10:00	04-11-2013, 02:12	1
12	04-11-2013, 12:00	2	05-11-2013, 12:00	07-11-2013, 10:00	07-11-2013, 10:00	08-11-2013, 08:00	1
13	06-11-2013, 13:48	3	07-11-2013, 15:48	09-11-2013, 10:00	09-11-2013, 10:00	10-11-2013, 02:12	2
14	07-11-2013, 12:00	4	08-11-2013, 12:00	10-11-2013, 10:00	10-11-2013, 10:00	11-11-2013, 08:00	3
15	09-11-2013, 18:00	5	10-11-2013, 03:00	11-11-2013, 10:00	11-11-2013, 10:00	12-11-2013, 08:00	1
16	14-11-2013, 21:48	1	15-11-2013, 07:48	16-11-2013, 10:00	16-11-2013, 10:00	17-11-2013, 02:12	1
17	16-11-2013, 09:48	2	17-11-2013, 13:48	19-11-2013, 10:00	19-11-2013, 10:00	20-11-2013, 02:12	2
18	14-11-2013, 20:00	3	16-11-2013, 16:00	19-11-2013, 10:00	19-11-2013, 10:00	20-11-2013, 08:00	3
19	19-11-2013, 22:00	4	20-11-2013, 05:00	21-11-2013, 10:00	21-11-2013, 10:00	22-11-2013, 08:00	1
20	24-11-2013, 09:48	5	25-11-2013, 13:48	27-11-2013, 10:00	27-11-2013, 10:00	28-11-2013, 02:12	1
21	22-11-2013, 20:00	1	24-11-2013, 16:00	27-11-2013, 10:00	27-11-2013, 10:00	28-11-2013, 08:00	2
22	27-11-2013, 22:00	2	28-11-2013, 05:00	29-11-2013, 10:00	29-11-2013, 10:00	30-11-2013, 08:00	3
23	29-11-2013, 21:48	3	30-11-2013, 19:48	02-12-2013, 10:00	02-12-2013, 10:00	03-12-2013, 02:12	1
24	01-12-2013, 12:00	4	02-12-2013, 12:00	04-12-2013, 10:00	04-12-2013, 10:00	05-12-2013, 08:00	2
25	05-12-2013, 17:48	5	06-12-2013, 05:48	07-12-2013, 10:00	07-12-2013, 10:00	08-12-2013, 02:12	1
26	06-12-2013, 14:00	1	07-12-2013, 13:00	09-12-2013, 10:00	09-12-2013, 10:00	10-12-2013, 08:00	2

**Gambar 4.11.** Hasil Simulasi Program berupa Penjadwalan untuk permintaan ke-11 sampai ke-26



Gambar 4.10 menunjukkan hasil simulasi dari fungsi *supplychain1to2* yang berupa jadwal untuk permintaan ke-1 sampai ke-16. Sedangkan Gambar 4.11 menunjukkan hasil simulasi untuk permintaan ke-11 sampai ke-26. Kolom pertama menunjukkan nomer urutan permintaan. Kolom kedua menunjukkan Jadwal mulai *loading*. Kolom ketiga menunjukkan urutan awal kapal tanker sebelum memprioritaskan pengiriman menggunakan kapal tanker yang telah kembali dari pengiriman sebelumnya. Kolom keempat berisi jadwal keberangkatan kapal tanker. Kolom kelima menunjukkan waktu selesainya proses *unloading* atau diterimanya solar di TBBM *customer*. Kolom keenam menunjukkan jam dan tanggal permintaan yang telah ditetapkan oleh TBBM *customer*. Jam dan tanggal pada kolom kelima dan keenam sama. Hal ini menunjukkan bahwa pengiriman solar menggunakan 5 kapal tanker dengan menggunakan jadwal *loading* yang diperoleh maka jam dan tanggal selesainya proses *unloading* atau diterimanya solar di TBBM *customer* sesuai dengan permintaan yang telah ditetapkan sebelumnya sehingga tidak terjadi kedatangan lebih awal atau keterlambatan.

Sedangkan kolom ketujuh menunjukkan waktu sampainya kapal tanker di TBBM Tuban setelah melakukan pengiriman ke TBBM *customer*. Kemudian dengan membandingkan waktu *loading* yang diperoleh dan waktu sampainya kapal tanker setelah melakukan pengiriman yang sebelumnya yaitu jika waktu sampainya kapal tanker setelah melakukan pengiriman yang sebelumnya kurang dari atau sama dengan waktu *loading* berikutnya maka kapal tanker yang digunakan untuk pengiriman sebelumnya bisa digunakan kembali untuk melakukan pengiriman berikutnya. Sehingga penggunaan kapal tanker dapat diatur kembali sedemikian seperti yang terdapat di kolom kedelapan.

Urutan baru kapal tanker pada kolom kedelapan menunjukkan bahwa 26 permintaan bisa dipenuhi oleh 4 kapal tanker saja dan tepat waktu seperti saat pengiriman menggunakan 5 kapal tanker. Dengan menggunakan 4 kapal tanker maka diperoleh pembagian sebagai berikut:

1. kapal 1 memenuhi permintaan ke-1, ke-2, ke-3, ke-4, ke-5, ke-9, ke-11, ke-12, ke-15, ke-16, ke-19, ke-20, ke-23 dan ke-25.
2. Kapal 2 memenuhi permintaan ke-6, ke-10, ke-13, ke-21, ke-24 dan ke-26.

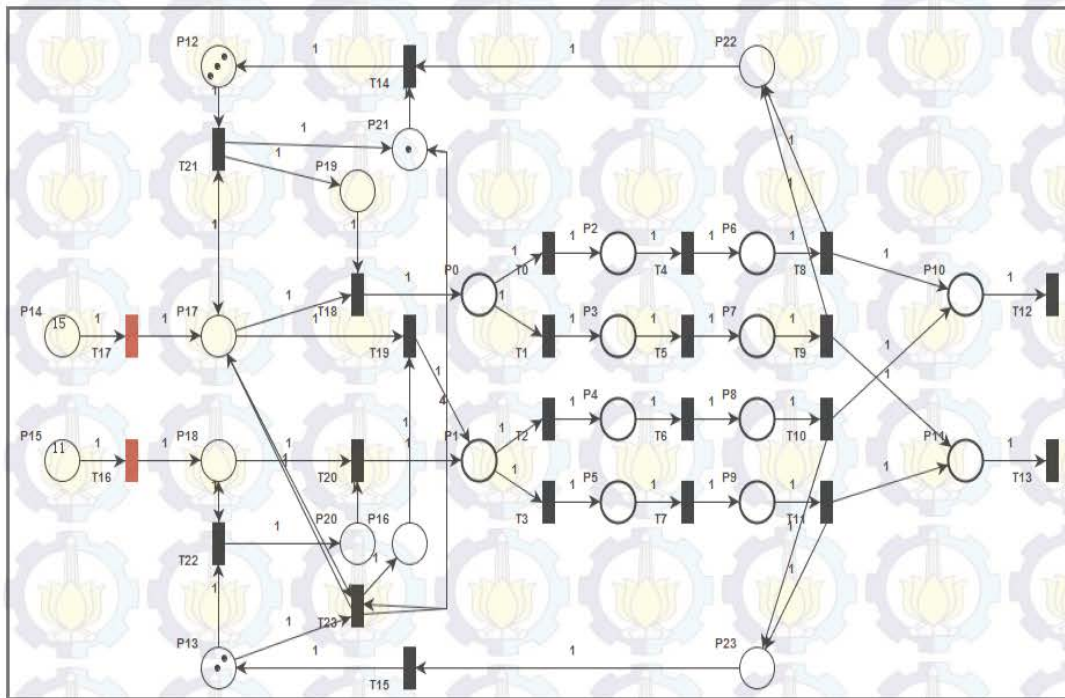
3. Kapal 3 memenuhi permintaan ke-7, ke-14, ke-18 dan ke-22.
4. Kapal 4 memenuhi permintaan ke-8.

#### **4.5 Model Rantai Pasok menggunakan *Petri Net* dengan mempertimbangkan perbedaan kapasitas Kapal Tanker**

Pada pembahasan ini dibuat model rantai pasok yang merupakan pengembangan dari model pertama yaitu untuk kemungkinan adanya volume permintaan sebesar 22.000 KI-25.000 KI sehingga kapasitas sebenarnya dari 5 kapal tanker dipertimbangkan. Model rantai pasok kedua ini dibuat menggunakan *Petri Net* dengan pengaturan prioritas kapal tanker untuk mengirim solar sesuai dengan kapasitasnya. Permintaan dengan volume kurang dari atau sama dengan 22.000 KI diprioritaskan menggunakan kapal tanker dengan kapasitas 22.000 KI. Sedangkan permintaan dengan volume sebesar 22.000 KI-25.000 KI menggunakan kapal tanker sebesar 25.000 KI. Dengan pengaturan prioritas ini diharapkan dapat meminimumkan kemungkinan terjadinya suatu keadaan ketika waktu *loading* seharusnya dimulai namun kapal tanker yang tersedia mempunyai kapasitas kurang dari volume permintaan yang ditetapkan oleh TBBM *customer*.

Model rantai pasok menggunakan *Petri Net* untuk volume permintaan tidak lebih dari 25.000 KI dibuat dengan 24 transisi dan 24 *place*. Terdapat 2 token pada *place*  $P_{13}$  yang menunjukkan jumlah kapal tanker dengan kapasitas 25.000 KI sedangkan jumlah kapal tanker dengan kapasitas 22.000 KI ditunjukkan oleh adanya 3 token pada *place*  $P_{12}$ . Pada model ini diasumsikan ada 15 permintaan dengan volume kurang dari atau sama dengan 22.000 KI yang ditunjukkan dengan adanya 15 token pada *place*  $P_{14}$ . Sedangkan permintaan dengan volume antara 22.000 KI-25.000 KI diasumsikan sebanyak 11 yang ditunjukkan dengan adanya 11 token pada *place*  $P_{15}$ . Pada *place*  $P_{21}$  memuat 1 token yang digunakan untuk mengatur prioritas penggunaan kapal tanker. Keadaan awal model *Petri Net* dengan volume permintaan kurang dari atau sama dengan 25.000 KI dapat dilihat pada Gambar 4.12.





**Gambar 4.12.** Keadaan awal *Petri Net* untuk Model Rantai Pasok dengan Volume sampai 25.000 K1

Keterangan:

*Place:*

$P_0$  = Proses *loading* pada kapal tanker berkapasitas 22.000 K1

$P_1$  = Proses *loading* pada kapal tanker berkapasitas 25.000 K1

$P_2$  = Perjalanan kapal tanker berkapasitas 22.000 K1 menuju TBBM Manggis

$P_3$  = Perjalanan kapal tanker berkapasitas 22.000 K1 menuju TBBM Tanjung Wangi

$P_4$  = Perjalanan kapal tanker berkapasitas 25.000 K1 menuju TBBM Manggis

$P_5$  = Perjalanan kapal tanker berkapasitas 25.000 K1 menuju TBBM Tanjung Wangi

$P_6$  = Proses *unloading* pada kapal tanker berkapasitas 22.000 K1 di TBBM Manggis

$P_7$  = Proses *unloading* pada kapal tanker berkapasitas 22.000 K1 di TBBM Tanjung Wangi

$P_8$  = Proses *unloading* pada kapal tanker berkapasitas 25.000 K1 di TBBM Manggis

$P_9$  = Proses *unloading* pada kapal tanker berkapasitas 22.000 K1 di TBBM Manggis

$P_{10}$  = Produk diterima di TBBM Manggis

$P_{11}$  = Produk diterima di TBBM Tanjung Wangi

$P_{12}$  = Kapal tanker berkapasitas 22.000 K1

$P_{13}$  = Kapal tanker berkapasitas 25.000 K1

$P_{14}$  = Produk solar untuk memenuhi permintaan sebesar kurang dari atau sama dengan 22.000 K1

$P_{15}$  = Produk solar untuk memenuhi permintaan sebesar 22.000 K1-25.000 K1

$P_{16}$  = Kapal tanker berkapasitas 25.000 K1 telah siap untuk mengirim solar sebesar kurang dari atau sama dengan 22.000 K1

$P_{17}$  = Solar dengan volume kurang dari atau sama dengan 22.000 K1 telah siap

$P_{18}$  = Solar dengan volume 22.000 K1-25.000 K1 telah siap

$P_{19}$  = Solar dengan volume kurang dari atau sama dengan 22.000 K1 mulai *loading* pada kapal tanker berkapasitas 22.000 K1

$P_{20}$  = Solar dengan volume 22.000 K1-25.000 K1 mulai *loading* pada kapal tanker berkapasitas 25.000 K1

$P_{21}$  = Ada tidaknya kapal tanker berkapasitas 22.000 K1 untuk memenuhi permintaan sebesar kurang dari atau sama dengan 22.000 K1

$P_{22}$  = Perjalanan kapal tanker berkapasitas 22.000 K1 menuju TBBM Tuban

$P_{23}$  = Perjalanan kapal tanker berkapasitas 23.000 K1 menuju TBBM Tuban

Transisi:

$T_0$  = Kapal tanker berkapasitas 22.000 K1 berangkat menuju TBBM Manggis

$T_1$  = Kapal tanker berkapasitas 22.000 K1 berangkat menuju TBBM Tanjung Wangi

$T_2$  = Kapal tanker berkapasitas 25.000 K1 berangkat menuju TBBM Manggis

$T_3$  = Kapal tanker berkapasitas 25.000 K1 berangkat menuju TBBM Tanjung Wangi

$T_4$  = Mulai proses *unloading* pada kapal tanker berkapasitas 22.000 K1 di TBBM Manggis



$T_5$  = Mulai proses *unloading* pada kapal tanker berkapasitas 22.000 K1 di TBBM Tanjung Wangi

$T_6$  = Mulai proses *unloading* pada kapal tanker berkapasitas 25.000 K1 di TBBM Manggis

$T_7$  = Mulai proses *unloading* pada kapal tanker berkapasitas 25.000 K1 di TBBM Tanjung Wangi

$T_8$  = Proses *unloading* pada kapal tanker berkapasitas 22.000 K1 di TBBM Manggis selesai

$T_9$  = Proses *unloading* pada kapal tanker berkapasitas 22.000 K1 di TBBM Tanjung Wangi selesai

$T_{10}$  = Proses *unloading* pada kapal tanker berkapasitas 25.000 K1 di TBBM Manggis selesai

$T_{11}$  = Proses *unloading* pada kapal tanker berkapasitas 25.000 K1 di TBBM Tanjung Wangi selesai

$T_{12}$  = Produk solar dengan volume kurang dari atau sama dengan 22.000 K1 siap dipasarkan

$T_{13}$  = Produk solar dengan volume sebesar 22.000 K1-25.000 K1 siap dipasarkan

$T_{14}$  = Kapal tanker berkapasitas 22.000 K1 masuk TBBM Tuban

$T_{15}$  = Kapal tanker berkapasitas 25.000 K1 masuk TBBM Tuban

$T_{16}$  = Solar untuk memenuhi permintaan sebesar 22.000 K1-25.000 K1 mulai disiapkan

$T_{17}$  = Solar untuk memenuhi permintaan sebesar kurang dari atau sama dengan 22.000 K1 mulai disiapkan

$T_{18}$  = Solar dengan volume kurang dari atau sama dengan 22.000 K1 mulai *loading* pada kapal tanker berkapasitas 22.000 K1

$T_{19}$  = Solar dengan volume kurang dari atau sama dengan 22.000 K1 mulai *loading* pada kapal tanker berkapasitas 25.000 K1

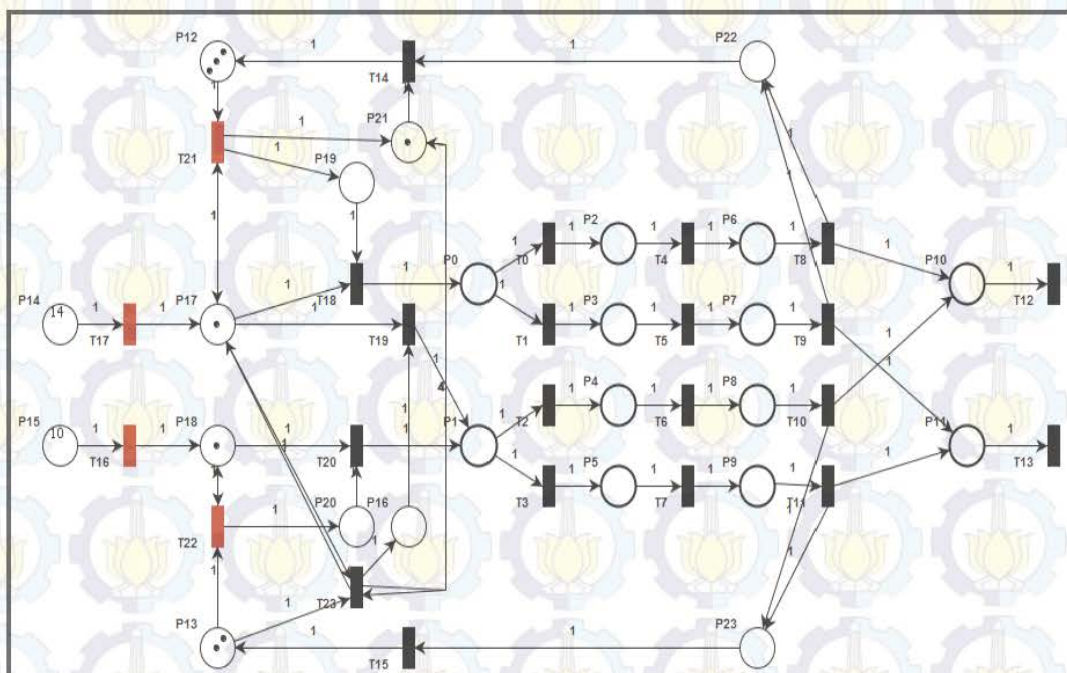
$T_{20}$  = Solar dengan volume 22.000 K1-25.000 K1 mulai *loading* pada kapal tanker berkapasitas 25.000 K1

$T_{21}$  = Kapal tanker berkapasitas 22.000 K1 mulai disiapkan

$T_{22}$  = Kapal tanker berkapasitas 25.000 KI untuk memenuhi permintaan sebesar kurang dari atau sama dengan 22.000 KI mulai disiapkan

$T_{23}$  = Kapal tanker berkapasitas 25.000 KI untuk memenuhi permintaan sebesar 22.000 KI-25.000KI mulai disiapkan

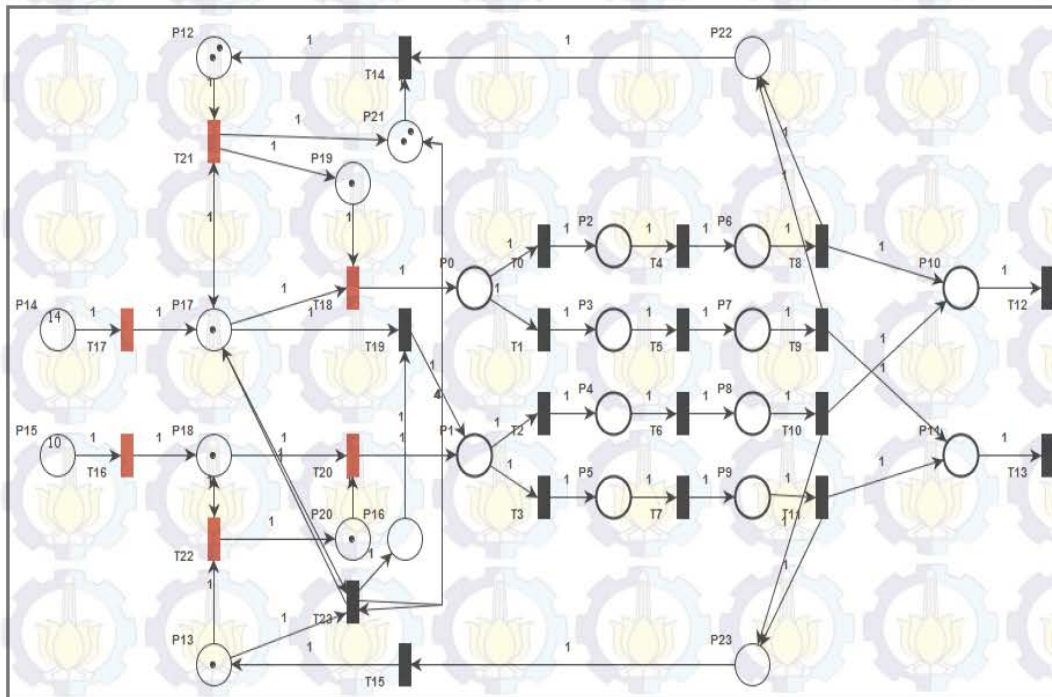
Gambar 4.12 menunjukkan keadaan awal *Petri Net* untuk model rantai pasok dengan mempertimbangkan perbedaan kapasitas kapal tanker. Transisi  $T_{16}$  dan  $T_{17}$  berwarna merah menunjukkan bahwa *enabled*. Transisi  $T_{16}$  *enabled* karena *place*  $P_{15}$  yang menjadi input dari transisi  $T_{16}$  mempunyai token lebih besar dari jumlah token minimum yang dibutuhkan oleh transisi  $T_{16}$ . Hal ini menunjukkan adanya solar untuk memenuhi volume permintaan sebesar 22.000 KI-25.000 KI. Transisi  $T_{17}$  *enabled* karena *place*  $P_{14}$  yang menjadi input dari transisi  $T_{17}$  mempunyai token lebih besar dari jumlah token minimum yang dibutuhkan oleh transisi  $T_{17}$ . Hal ini menunjukkan adanya solar untuk memenuhi volume permintaan sebesar kurang dari atau sama dengan 22.000. Selanjutnya pada Gambar 4.13 menunjukkan keadaan setelah  $T_{16}$  dan  $T_{17}$  *difire* masing-masing sebanyak 1 kali.



**Gambar 4.13.** Keadaan setelah Transisi  $T_{16}$  dan  $T_{17}$  *Difire* sebanyak 1 Kali



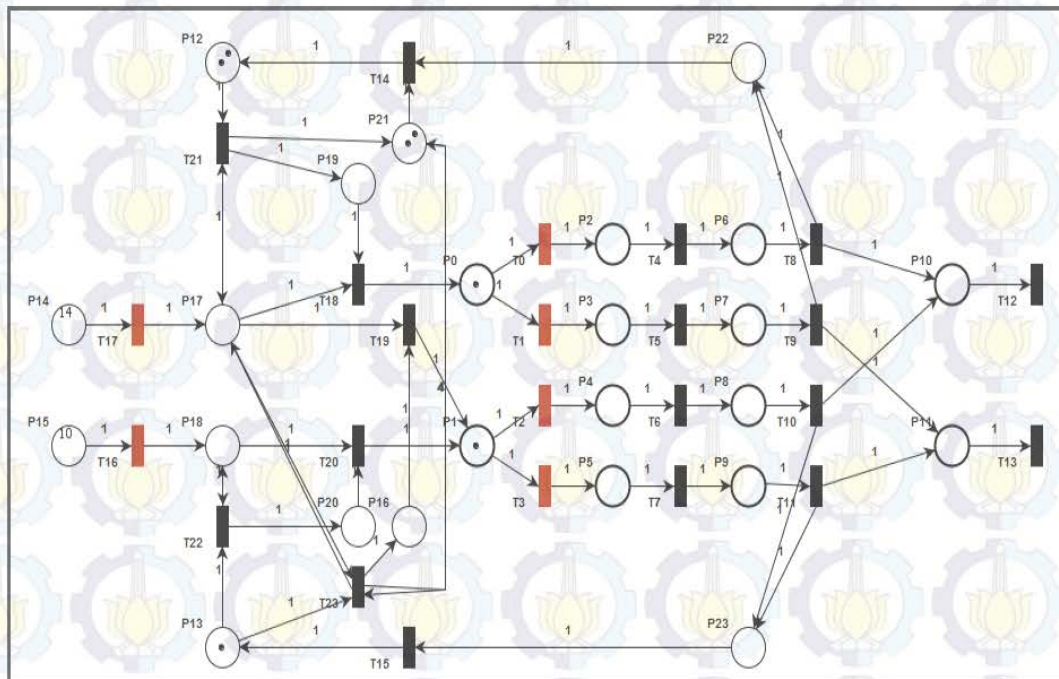
Setelah transisi  $T_{16}$  *difire* maka transisi  $T_{21}$  menjadi *enabled*. Hal ini menunjukkan bahwa kapal tanker dengan kapasitas 22.000 Kl bisa dipersiapkan untuk melakukan pengiriman dengan volume permintaan sebesar kurang dari atau sama dengan 22.000 Kl setelah produk yang akan dikirim telah siap. Transisi  $T_{22}$  menjadi *enabled* setelah transisi  $T_{17}$  *difire* menunjukkan bahwa kapal tanker dengan kapasitas 25.000 Kl bisa dipersiapkan untuk melakukan pengiriman dengan volume permintaan sebesar 22.000 Kl-25.000 Kl setelah produk yang akan dikirim telah siap. Sedangkan transisi  $T_{23}$  tetap tidak *enabled*. Hal ini menunjukkan bahwa kapal tanker dengan kapasitas 25.000 Kl tidak dapat dipersiapkan untuk melakukan pengiriman dengan volume permintaan sebesar kurang dari atau sama dengan 22.000 Kl karena pengiriman diprioritaskan menggunakan kapal tanker dengan kapasitas 22.000 Kl. Selanjutnya keadaan setelah transisi  $T_{21}$  dan  $T_{22}$  *difire* ditunjukkan pada Gambar 4.14.



**Gambar 4.14.** Keadaan setelah Transisi  $T_{21}$  dan  $T_{22}$  *Difire* sebanyak 1 Kali

Gambar 4.14 menunjukkan bahwa sebuah kapal tanker dengan kapasitas 22.000 Kl telah siap untuk mengirim solar dengan volume kurang dari atau sama

dengan 22.000 KI dan sebuah kapal tanker dengan kapasitas 25.000 KI telah siap untuk mengirim solar dengan volume 22.000 KI-25.000 KI. Kapal tanker yang siap ditunjukkan dengan adanya masing-masing 1 token pada *place*  $P_{19}$  dan  $P_{20}$ . Setelah transisi  $T_{21}$  dan  $T_{22}$  *difire* maka transisi  $T_{18}$  dan  $T_{19}$  menjadi *enabled*. Hal ini menunjukkan bahwa proses *loading* dengan kapal tanker berkapasitas 22.000 KI dan 25.000 KI siap dilakukan. Selanjutnya setelah transisi  $T_{18}$  *difire* menunjukkan keadaan setelah proses *loading* menggunakan kapal tanker dengan kapasitas 22.000 KI telah dimulai dan setelah transisi  $T_{19}$  *difire* menunjukkan keadaan setelah proses *loading* menggunakan kapal tanker dengan kapasitas 25.000 KI telah dimulai. Keadaan setelah transisi  $T_{18}$  dan  $T_{19}$  *difire* ditunjukkan pada Gambar 4.15.

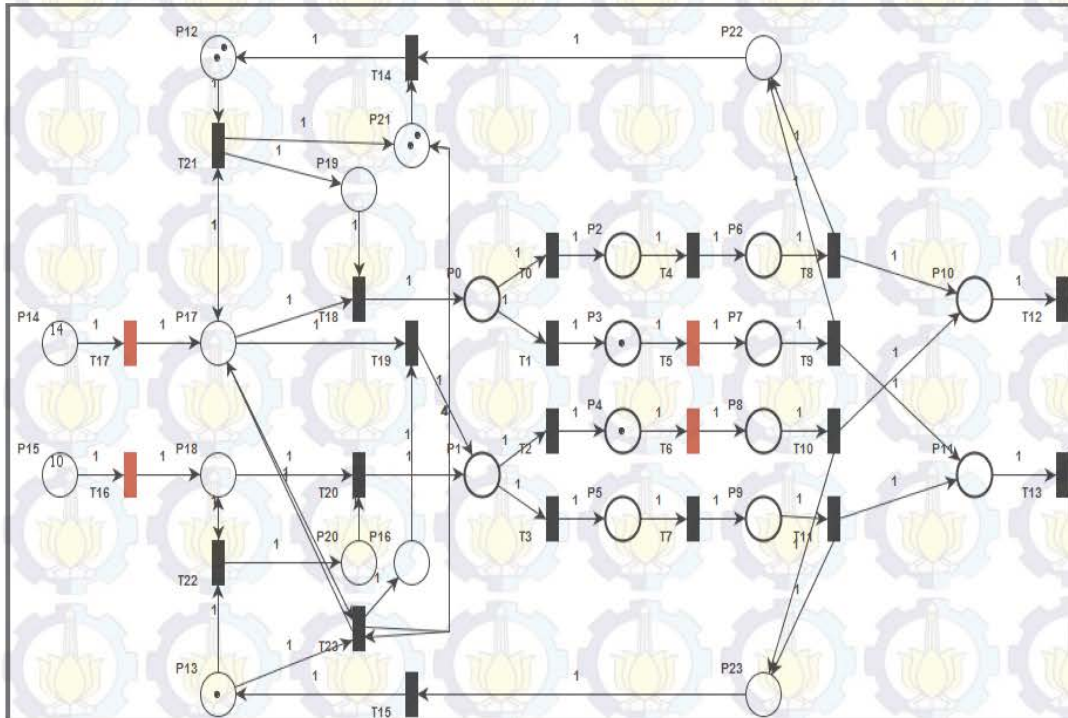


**Gambar 4.15.** Keadaan setelah Transisi  $T_{18}$  dan  $T_{19}$  *Difire*

Gambar 4.15 menunjukkan bahwa proses *loading* telah selesai dan produk siap untuk dikirimkan ke TBBM Tanjung Wangi atau TBBM Manggis. Hal ini ditunjukkan dengan warna merah pada transisi  $T_0$  dan  $T_1$  untuk pengiriman menggunakan kapal tanker berkapasitas 22.000 KI dan warna merah pada transisi  $T_2$  dan  $T_3$  untuk pengiriman menggunakan kapal tanker berkapasitas 25.000 KI.

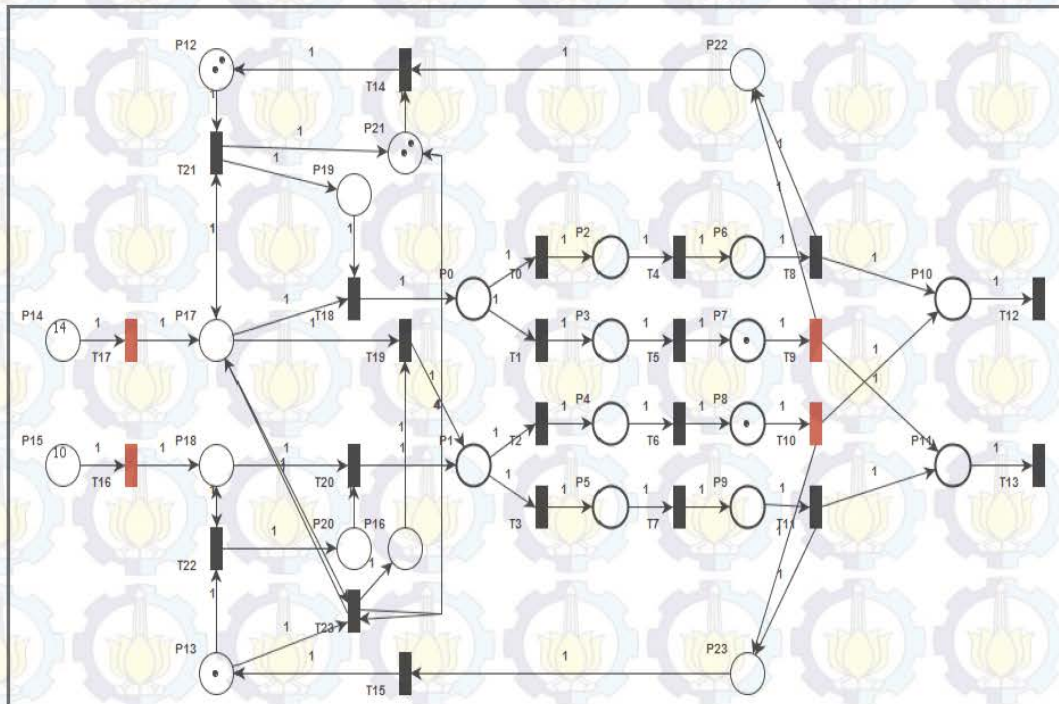


Selanjutnya ketika solar dikirim menggunakan kapal tanker berkapasitas 22.000 Kl ke TBBM Tanjung Wangi maka transisi  $T_1$  yang *difire*. Sedangkan jika solar akan dikirim menggunakan kapal tanker berkapasitas 25.000 Kl ke TBBM Manggis maka transisi  $T_2$  yang *difire*. Keadaan ketika transisi  $T_1$  dan  $T_2$  telah *difire* ditunjukkan pada Gambar 4.16.



**Gambar 4.16.** Keadaan setelah Transisi  $T_1$  dan  $T_2$  *Difire*

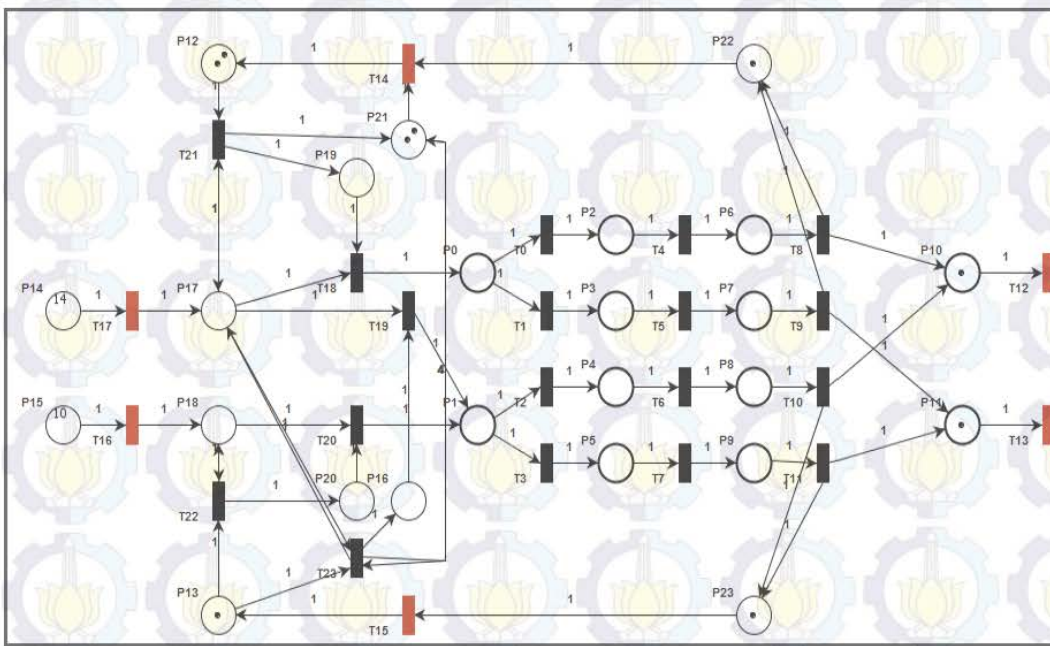
Gambar 4.16 menunjukkan bahwa sebuah kapal tanker berangkat ke TBBM Tanjung Wangi untuk memenuhi permintaan dengan volume kurang dari atau sama dengan 22.000 Kl dan sebuah kapal tanker lainnya berangkat ke TBBM Manggis untuk memenuhi permintaan dengan volume 22.000 Kl-25.000 Kl. Setelah transisi  $T_1$  *difire* maka transisi  $T_0$  menjadi tidak *enabled* dan setelah transisi  $T_2$  *difire* maka transisi  $T_3$  menjadi tidak *enabled*. Hal ini menunjukkan sebuah kapal tanker hanya bisa berangkat ke satu tempat pada saat tertentu dan tidak mungkin ke dua tempat dalam waktu bersamaan. Selanjutnya setelah transisi  $T_1$  dan  $T_2$  *difire* maka transisi  $T_5$  dan  $T_6$  menjadi *enabled*. Keadaan setelah transisi  $T_5$  dan  $T_6$  *difire* ditunjukkan pada Gambar 4.17.



**Gambar 4.17.** Keadaan setelah Transisi  $T_5$  dan  $T_6$  Difire

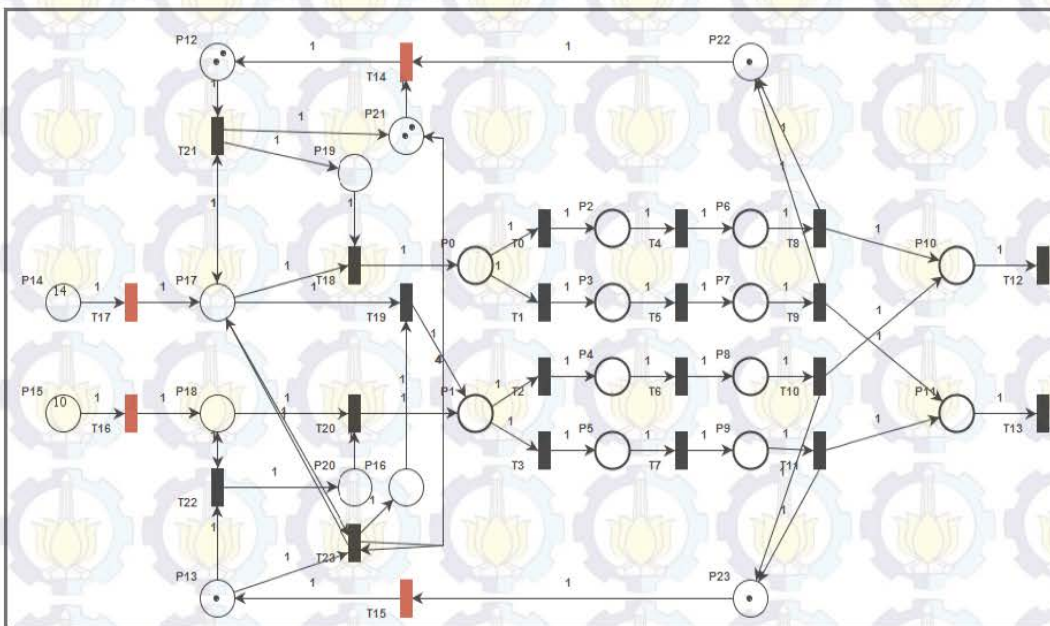
Gambar 4.17 menunjukkan bahwa setelah transisi  $T_5$  maka  $T_9$  menjadi *enabled* dan setelah transisi  $T_6$  difire maka transisi  $T_{10}$  menjadi *enabled*. Hal ini menunjukkan bahwa kapal tanker dengan kapasitas 25.000 Kl telah sampai di TBBM Tanjung Wangi dan kapal tanker dengan kapasitas 22.000 Kl telah sampai di TBBM Manggis. proses *unloading* solar dengan volume sebesar kurang dari atau sama dengan 22.000 Kl di TBBM Tanjung Wangi dan volume sebesar 22.000 Kl-25.000 Kl di TBBM Manggis telah dimulai. Setelah proses *unloading* selesai maka produk diterima oleh masing-masing TBBM *customer* tersebut. Hal ini ditunjukkan pada Gambar 4.18 yaitu keadaan setelah transisi  $T_9$  dan  $T_{10}$  difire.





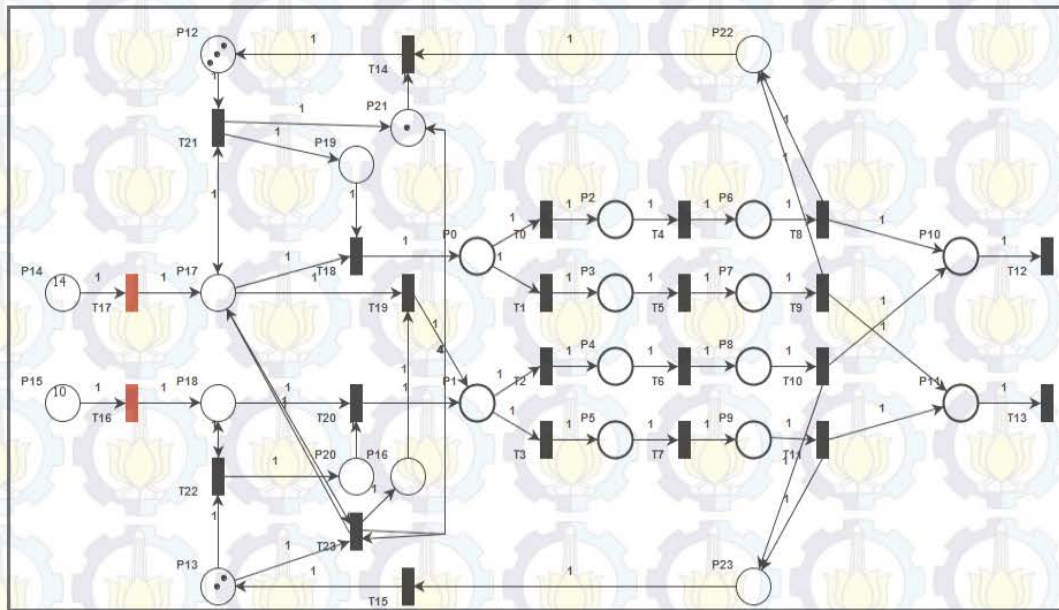
**Gambar 4.18.** Keadaan setelah Transisi  $T_9$  dan  $T_{10}$  Difire

Selanjutnya keadaan ketika masing- masing produk telah masuk di gudang TBBM Manggis dan Tanjung Wangi ditunjukkan oleh Gambar 4.19. Gambar 4.19 menunjukkan keadaan ketika transisi  $T_{12}$  dan  $T_{13}$  telah difire.



**Gambar 4.19.** Keadaan setelah Transisi  $T_{12}$  dan  $T_{13}$  Difire

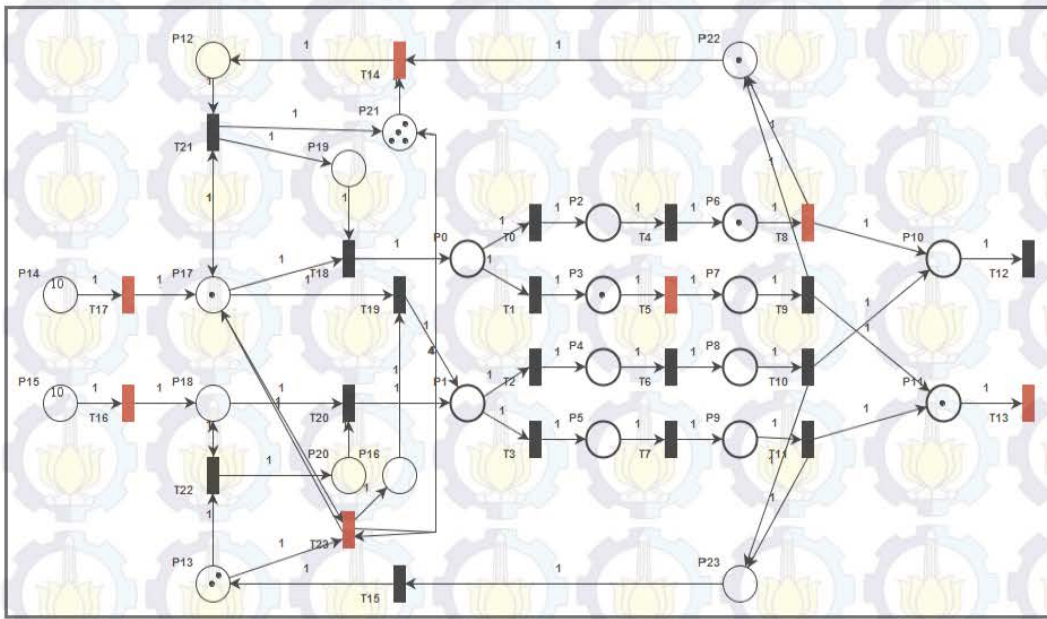
Keadaan ketika transisi  $T_{14}$  telah *difire* menunjukkan bahwa kapal tanker dengan kapasitas 22.000 KI telah kembali ke TBBM Tuban. Sedangkan ketika transisi  $T_{15}$  telah *difire* menunjukkan bahwa kapal tanker dengan kapasitas 25.000 KI telah kembali ke TBBM Tuban. Keadaan ini ditunjukkan pada Gambar 4.20.



**Gambar 4.20.** Keadaan setelah Transisi  $T_{14}$  dan  $T_{15}$  *Difire*

Selanjutnya keadaan ketika kapal tanker dengan kapasitas 22.000 KI belum kembali dari perjalanan sedangkan pengiriman untuk volume permintaan sebesar kurang dari atau sama dengan 22.000 KI akan segera dilakukan maka bisa menggunakan kapal tanker dengan kapasitas 25.000 KI. Hal ini ditunjukkan pada Gambar 4.21.





**Gambar 4.21.** Keadaan ketika Transisi  $T_{23}$  menjadi *Enabled*

Gambar 4.21 menunjukkan bahwa 3 kapal tanker dengan kapasitas 22.000 Kl belum kembali dari pengiriman sebelumnya. Satu kapal tanker sedang dalam perjalanan menuju TBBM Tanjung Wangi, satu kapal tanker sedang dalam proses *unloading* di TBBM Manggis dan satu kapal tanker lainnya sedang perjalanan kembali dari TBBM Tanjung Wangi. Ketika pengiriman selanjutnya dengan volume kurang dari atau sama dengan 22.000 Kl harus segera dilakukan maka bisa menggunakan kapal tanker dengan kapasitas 25.000 Kl yang ditunjukkan dengan transisi  $T_{23}$  yang menjadi *enabled*. Transisi  $T_{23}$  menjadi *enabled* hanya jika transisi  $T_{17}$  telah *fire* sedangkan pada *place*  $P_{12}$  tidak ada token.

#### 4.6 Model Aljabar *Max Plus* dari Model *Petri Net* dengan mempertimbangkan perbedaan kapasitas Kapal Tanker

Model *Petri Net* yang telah dibuat untuk kemungkinan adanya volume permintaan sebesar 22.000 Kl-25.000 Kl selanjutnya dijabarkan dalam model Aljabar *Max Plus*. Untuk volume permintaan kurang dari atau sama dengan 22.000 Kl maka bisa menggunakan kapal tanker dengan kapasitas 22.000 Kl atau 25.000 Kl. Namun dalam model ini diprioritaskan menggunakan kapal tanker dengan kapasitas 22.000 Kl. Sedangkan untuk volume permintaan sebesar 22.000

KI-25.000 KI maka hanya bisa dipenuhi menggunakan kapal tanker dengan kapasitas 25.000 KI.

Nilai  $m$  menyatakan jumlah kapal tanker yang dapat digunakan untuk melakukan pengiriman solar. Sehingga untuk kapal tanker dengan kapasitas 22.000 KI maka nilai  $m = 3$  dan untuk kapal tanker dengan kapasitas 25.000 KI maka nilai  $m = 2$ . sedangkan  $k$  menyatakan urutan permintaan.

Berdasarkan model *Petri Net* yang telah dibentuk menunjukkan bahwa untuk permintaan ke-1 sampai ke- $m$  dengan  $m$  merupakan banyaknya kapal tanker yang bisa digunakan maka waktu *loading* langsung bisa segera dilakukan setelah produk solar siap karena kapal tanker telah tersedia. Sedangkan waktu *loading* untuk permintaan berikutnya harus menunggu kapal tanker kembali dari pengiriman sehingga waktu *loading* juga dipengaruhi oleh sampainya kapal tanker di TBBM *customer* dan lamanya perjalanan dari TBBM *customer* pada permintaan sebelumnya. Waktu sampainya produk di TBBM *customer* merupakan waktu selesainya proses *unloading* sehingga dipengaruhi oleh waktu *loading*, lama proses *loading*, lama keberangkatan menuju TBBM *customer* dan lama proses *unloading*. Dan waktu diterimanya produk solar di TBBM *customer* sesuai dengan waktu selesainya proses *unloading*. Berdasarkan kriteria tersebut maka diperoleh model Aljabar *Max Plus* untuk model kedua sebagai berikut:

$$t_{xc}(k) = \begin{cases} t_{yc}(k-m) \otimes p_{ic}(k-m) \oplus u_c(k), & \text{untuk } k > m \\ u_c(k), & \text{untuk } 1 \leq k \leq m \end{cases} \quad (4.23)$$

$$t_{yc}(k) = t_{xc}(k) \otimes p_{fc}(k) \otimes p_{gc}(k) \otimes p_{hc}(k) \quad (4.24)$$

$$y_c(k) = t_{yc}(k) \quad (4.25)$$

Keterangan:

$t_{xc}(k)$  = waktu *loading* permintaan ke- $k$

$t_{yc}(k)$  = waktu sampai permintaan ke- $k$

$y_c(k)$  = waktu produk diterima saat ke- $k$

$p_{fc}(k)$  = lama *loading* permintaan ke- $k$

$p_{gc}(k)$  = lama berangkat permintaan ke- $k$

$p_{hc}(k)$  = lama *unloading* permintaan ke- $k$



$p_{ic}(k)$  = lama kembali saat ke- $k$

$u_c(k)$  = waktu masuknya solar saat ke- $k$

Selanjutnya dengan memisalkan:

$$t_c(k) = \begin{bmatrix} t_{xc}(k) \\ t_{yc}(k) \end{bmatrix}, A_{0c}(k) = \begin{bmatrix} p_{fc}(k) \otimes p_{gc}(k) \otimes p_{hc}(k) & \varepsilon \\ \varepsilon & \varepsilon \end{bmatrix},$$

$$A_{1c}(k) = \begin{bmatrix} \varepsilon & p_i(k-m) \\ \varepsilon & \varepsilon \end{bmatrix}, B_{0c} = \begin{bmatrix} e \\ \varepsilon \end{bmatrix}, \text{ dan } J = \begin{bmatrix} \varepsilon & e \end{bmatrix}, \text{ dengan } e = 0$$

Maka Persamaan (4.23) dan (4.25) dapat ditulis menjadi Persamaan (4.27) sebagai berikut:

$$t_c(k) = A_{0c}(k) \otimes t_c(k) \oplus A_{1c}(k) \otimes t_c(k-m) \oplus B_{0c} \otimes u_c(k) \quad (4.27)$$

Substitusi  $t_c(k)$  pada ruas kiri Persamaan (4.27) ke ruas kanannya sehingga diperoleh persamaan sebagai berikut:

$$t_c(k) = A_{0c}^2(k) \otimes t_c(k) \oplus (E \oplus A_{0c}(k)) \otimes A_{1c}(k) \otimes t_c(k-m) \oplus (E \oplus A_{0c}(k)) \otimes B_{0c} \otimes u_c(k) \quad (4.28)$$

dengan  $E = \begin{bmatrix} 0 & \varepsilon \\ \varepsilon & 0 \end{bmatrix}$  yang merupakan matriks identitas berukuran 2x2.

Kemudian substitusikan kembali  $t_c(k)$  pada ruas kiri Persamaan (4.28) ke ruas kanannya sehingga diperoleh persamaan sebagai berikut:

$$t_c(k) = A_{0c}^3(k) \otimes t_c(k) \oplus (E \oplus A_{0c}(k) \oplus A_{0c}^2(k)) \otimes A_{1c}(k) \otimes t_c(k-m) \oplus (E \oplus A_{0c}(k) \oplus A_{0c}^2(k)) \otimes B_{0c} \otimes u_c(k)$$

Ulangi lagi substitusi sebanyak  $\beta$  sehingga diperoleh persamaan sebagai berikut:

$$t_c(k) = A_{0c}^\beta(k) \otimes t_c(k) \oplus (E \oplus A_{0c}(k) \oplus \dots \oplus A_{0c}^{\beta-1}(k)) \otimes A_{1c}(k) \otimes t_c(k-m) \oplus (E \oplus A_{0c}(k) \oplus \dots \oplus A_{0c}^{\beta-1}(k)) \otimes B_{0c} \otimes u_c(k) \quad (4.29)$$

Lalu untuk  $\beta$  mendekati tak hingga maka Persamaan (4.29) menjadi persamaan baru sebagai berikut:

$$t_c(k) = (E \oplus A_{0c}(k) \oplus A_{0c}^2(k) \oplus \dots) \otimes A_{1c}(k) \otimes t_c(k-m) \oplus (E \oplus A_{0c}(k) \oplus A_{0c}^2(k) \oplus \dots) \otimes B_{0c} \otimes u_c(k) \quad (4.30)$$

Berdasarkan Definisi 2.10 maka Persamaan (4.30) dapat ditulis menjadi Persamaan (4.31) sebagai berikut:

$$t_c(k) = A_{0c}^*(k) \otimes A_{1c}(k) \otimes t_c(k-m) \oplus A_{0c}^*(k) \otimes B_{0c} \otimes u_c(k) \quad (4.31)$$

Selanjutnya Persamaan (4.31) dapat ditulis menjadi Persamaan (4.32) sebagai berikut:

$$t_c(k) = A_c(k) \otimes t_c(k-m) \oplus B_c(k) \otimes u_c(k) \quad (4.32)$$

dengan:

$$A_c(k) = A_{0c}^*(k) \otimes A_{1c}(k) \text{ dan}$$

$$B_c(k) = A_{0c}^*(k) \otimes B_{0c}$$

Sedangkan Persamaan (4.26) dapat ditulis menjadi Persamaan (4.33) sebagai berikut:

$$y_c(k) = J \otimes t_c(k) \quad (4.33)$$

Selanjutnya dengan mensubstitusi  $t_c(k)$  yang terdapat pada Persamaan (4.32) ke dalam Persamaan (4.33) maka diperoleh persamaan sebagai berikut:

$$y_c(k) = J \otimes A_c(k) \otimes t_c(k-m) \oplus J \otimes B_c(k) \otimes u_c(k) \quad (4.34)$$

Kemudian substitusikan kembali  $t_c(k)$  pada Persamaan (4.32) Persamaan (4.34) sehingga diperoleh persamaan sebagai berikut:

$$y_c(k) = J \otimes A_c(k) \otimes A_c(k-m) \otimes t_c(k-2m) \oplus J \otimes A_c(k) \otimes B_c(k-m) \otimes u_c(k-m) \oplus J \otimes B_c(k) \otimes u_c(k)$$

Ulangi lagi substitusi sebanyak  $\alpha$  sehingga diperoleh persamaan sebagai berikut:

$$y_c(k) = [J \otimes (A_c(k) \otimes A_c(k-m) \otimes \dots \otimes A_c(k-(\alpha-1)m)) \otimes t_c(k-\alpha m)] \oplus [J \otimes B_c(k) \otimes u_c(k)] \oplus [J \otimes A_c(k) \otimes B_c(k-m) \otimes u_c(k-m)] \oplus [J \otimes A_c(k) \otimes A_c(k-m) \otimes B_c(k-2m) \otimes u_c(k-2m)] \oplus \dots \oplus [J \otimes (A_c(k) \otimes A_c(k-m) \otimes \dots \otimes A_c(k-(\alpha-2)m)) \otimes B_c(k-(\alpha-1)m) \otimes u_c(k-(\alpha-1)m)] \quad (4.35)$$

Kemudian untuk  $k > m$  maka Persamaan (4.24) dan (4.26) dapat ditulis menjadi Persamaan (4.35) sebagai berikut:

$$t_c(k) = A_{0c}(k) \otimes t_c(k) \oplus B_{0c} \otimes u_c(k) \quad (4.36)$$

Substitusi  $t_c(k)$  pada ruas kiri Persamaan (4.36) ke ruas kanannya sehingga diperoleh persamaan sebagai berikut:

$$t_c(k) = A_{0c}^2(k) \otimes t_c(k) \oplus A_{0c}(k) \otimes B_{0c} \otimes u_c(k) \oplus B_{0c} \otimes u_c(k) \quad (4.37)$$



Kemudian substitusikan kembali  $t_c(k)$  pada ruas kiri Persamaan (4.37) ke ruas kanannya sehingga diperoleh persamaan sebagai berikut:

$$t_c(k) = A_{0c}^3(k) \otimes t_c(k) \oplus A_{0c}^2(k) \otimes B_{0c} \otimes u_c(k) \oplus A_{0c}(k) \otimes B_{0c} \otimes u_c(k) \oplus B_{0c} \otimes u_c(k)$$

Ulangi lagi substitusi sebanyak  $\beta$  sehingga diperoleh persamaan sebagai berikut:

$$t_c(k) = A_{0c}^\beta(k) \otimes t_c(k) \oplus (E \oplus A_{0c}(k) \oplus \dots \oplus A_{0c}^{\beta-1}(k)) \otimes B_{0c} \otimes u_c(k) \quad (4.38)$$

Lalu untuk  $\beta$  mendekati tak hingga maka Persamaan (4.38) menjadi persamaan baru sebagai berikut:

$$t_c(k) = (E \oplus A_{0c}(k) \oplus A_{0c}^2(k) \oplus \dots) \otimes B_{0c} \otimes u_c(k) \quad (4.38)$$

Berdasarkan Definisi 2.10 maka Persamaan (4.38) dapat ditulis menjadi Persamaan (4.39) sebagai berikut:

$$t_c = A_{0c}^*(k) \otimes B_{0c} \otimes u_c(k) \quad (4.39)$$

Selanjutnya Persamaan (4.39) dapat ditulis menjadi Persamaan (4.40) sebagai berikut:

$$t_c(k) = B_c(k) \otimes u_c(k) \quad (4.40)$$

dengan:

$$B_c(k) = A_{0c}^*(k) \otimes B_{0c}$$

Berdasarkan Persamaan (4.39) maka diperoleh persamaan sebagai berikut:

$$t_c(k - \alpha m) = B_c(k - \alpha m) \otimes u_c(k - \alpha m) \quad (4.41)$$

Selanjutnya substitusi Persamaan (4.41) ke Persamaan (4.35) sehingga diperoleh persamaan baru sebagai berikut:

$$\begin{aligned} y_c(k) = & [J \otimes B_c(k) \otimes u_c(k)] \oplus [J \otimes A_c(k) \otimes B_c(k - m) \otimes u_c(k - m)] \oplus [J \otimes A_c(k) \\ & \otimes A_c(k - m) \otimes B_c(k - 2m) \otimes u_c(k - 2m)] \oplus \dots \oplus [J \otimes (A_c(k) \otimes \\ & A_c(k - m) \otimes \dots \otimes A_c(k - (\alpha - 2)m)) \otimes B_c(k - (\alpha - 1)m) \otimes \\ & u_c(k - (\alpha - 1)m)] \oplus [J \otimes (A_c(k) \otimes A_c(k - m) \otimes \dots \otimes \\ & A_c(k - (\alpha - 1)m)) \otimes B_c(k - \alpha m) \otimes u_c(k - \alpha m)] \end{aligned} \quad (4.42)$$

Selanjutnya berdasarkan Persamaan 4.43 maka dapat dibentuk persamaan sebagai berikut:

$$Y_c = H_c \otimes U_c \quad (4.43)$$

dengan elemen-elemen pada matriks  $Y_c$  diperoleh dari data waktu permintaan yang telah ditetapkan oleh TBBM *customer*, elemen-elemen pada matriks  $H_c$



diperoleh berdasarkan hasil perhitungan data-data sekunder yang telah terkumpul dan elemen-elemen matrik  $U_c$  merupakan waktu mulai *loading* yang akan dihitung.

Untuk banyaknya permintaan dinyatakan dengan  $m_d$ , banyaknya kapal tanker dinyatakan dengan  $m$  dan  $x = s + \gamma m$  menyatakan urutan permintaan dimulai dari 0 dengan  $\gamma$  merupakan hasil bagi  $x$  terhadap  $m$  dan  $s$  merupakan sisa pembagian  $x$  terhadap  $m$  atau  $x$  modulo  $m$ , maka diperoleh penjabaran bentuk matriks  $Y_c, H_c$  dan  $U_c$  sebagai berikut:

$$Y_c = \begin{bmatrix} y(s) \\ y(s+n) \\ y(s+2n) \\ \vdots \\ y(s+\gamma n) \end{bmatrix}, U_c = \begin{bmatrix} u(s) \\ u(s+n) \\ u(s+2n) \\ \vdots \\ u(s+\gamma n) \end{bmatrix}, \text{ dan}$$

$$H = \begin{bmatrix} CB(s) & \varepsilon & \dots & \varepsilon \\ CA(s+n)B(s) & CB(s+n) & \dots & \varepsilon \\ \vdots & \vdots & \ddots & \vdots \\ CA(s+n)A(s+2n) \dots A(s+\gamma n)B(s) & \dots & \dots & CB(s+\gamma n) \end{bmatrix}$$

Keterangan:

$$\begin{aligned} A(s+n) &= A_0^*(s+n) \otimes A_1(s+n) \\ &= (E \oplus A_0(s+n) \oplus A_0^2(s+n) \oplus \dots) \otimes A_1(s+n) \\ &= \left( \begin{bmatrix} e & \varepsilon \\ \varepsilon & e \end{bmatrix} \oplus \begin{bmatrix} \varepsilon & \varepsilon \\ p_a(s+n) \otimes p_b(s+n) \otimes p_c(s+n) & \varepsilon \end{bmatrix} \oplus \dots \right) \otimes \begin{bmatrix} \varepsilon & p_d(l) \\ \varepsilon & \varepsilon \end{bmatrix} \\ &= \begin{bmatrix} \varepsilon & p_d(s) \\ \varepsilon & p_a(s+n) \otimes p_b(s+n) \otimes p_c(s+n) \end{bmatrix} \end{aligned}$$

$$\begin{aligned} B(s) &= A_0^*(s) \otimes B_0 \\ &= \begin{bmatrix} e & \varepsilon \\ p_a(s) \otimes p_b(s) \otimes p_c(s) & e \end{bmatrix} \otimes \begin{bmatrix} e \\ \varepsilon \end{bmatrix} \\ &= \begin{bmatrix} e \\ p_a(s) \otimes p_b(s) \otimes p_c(s) \end{bmatrix} \end{aligned}$$

Pada penelitian ini untuk volume tidak melebihi 22.000 KI jumlah permintaannya sebanyak 15 sehingga  $m_d = 15$ . Pengiriman diprioritaskan menggunakan kapal tanker berkapasitas 22.000 KI yang jumlahnya ada 3 sehingga  $m = 3$  dan  $l = 0,1,2$ . Urutan permintaan adalah  $x = 0,1,2, \dots, 14$  dengan  $x = 0$  menunjukkan permintaan ke-1 untuk volume tidak melebihi 22.000 KI,  $x = 1$  menunjukkan permintaan ke-2 untuk volume tidak melebihi 22.000 KI, dan

seterusnya sampai  $x = 14$  menunjukkan permintaan ke-15 untuk volume tidak melebihi 22.000 Kl. Sedangkan  $\gamma$  merupakan hasil bagi  $x$  terhadap 3 dan  $s = 0,1,2$  merupakan sisa pembagian  $x$  terhadap 3 atau  $x$  modulo 3. Matriks  $Y_c$  dan  $H_c$  untuk volume tidak melebihi 22.000 Kl dinotasikan dengan  $Y_b$  dan  $H_b$ . Selanjutnya diperoleh matriks  $Y_b$  dalam urutan permintaan yang sebenarnya ( $k$ ) dan dalam  $x$  sebagai berikut:

$$Y_b(k) \rightarrow Y_b(x)$$

$$\begin{bmatrix} y(1) \\ y(2) \\ y(3) \\ y(4) \\ y(5) \\ y(6) \\ y(7) \\ y(8) \\ y(9) \\ y(10) \\ y(11) \\ y(12) \\ y(13) \\ y(14) \\ y(15) \end{bmatrix} \rightarrow \begin{bmatrix} y(0) \\ y(1) \\ y(2) \\ y(3) \\ y(4) \\ y(5) \\ y(6) \\ y(7) \\ y(8) \\ y(9) \\ y(10) \\ y(11) \\ y(12) \\ y(13) \\ y(14) \end{bmatrix} = \begin{bmatrix} y(0) \\ y(1) \\ y(2) \\ y(0+1.3) \\ y(1+1.3) \\ y(2+1.3) \\ y(0+2.3) \\ y(1+2.3) \\ y(2+2.3) \\ y(0+3.3) \\ y(1+3.3) \\ y(2+3.3) \\ y(0+4.3) \\ y(1+4.3) \\ y(2+4.3) \end{bmatrix}$$

Untuk  $x$  modulo 5 = 0 atau  $s = 0$  menunjukkan bahwa permintaan dipenuhi oleh kapal tanker ke-1 (berkapasitas 22.000 Kl). maka  $x = 3.n$  dengan  $n = 0,1,2,3,4$  sehingga  $k = x + 1 = 3.n+1=3.n + 1$  dan diperoleh:

$$Y_b 1 = \begin{bmatrix} y(1) \\ y(4) \\ y(7) \\ y(10) \\ y(13) \end{bmatrix} \text{ dan}$$

$$H_b 1 = \begin{bmatrix} CB(1) & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ CA(4)B(1) & CB(4) & \varepsilon & \varepsilon & \varepsilon \\ CA(7)A(4)B(1) & CA(7)B(4) & CB(7) & \varepsilon & \varepsilon \\ CA(10)A(7)A(4)B(1) & CA(10)A(7)B(4) & CA(10)B(7) & CB(10) & \varepsilon \\ CA(13)A(10)A(7)A(4)B(1) & CA(13)A(10)A(7)B(4) & CA(13)A(10)B(7) & CA(13)B(10) & CB(13) \end{bmatrix}$$



Untuk  $x$  modulo 3 = 1 atau  $s = 1$  menunjukkan bahwa permintaan dipenuhi oleh kapal tanker ke-2 (berkapasitas 22.000 Kl). maka  $x = 3.n + 1$  dengan  $n = 0,1,2,3,4$  sehingga  $k = x + 1 = 3.n + 1 + 1 = 3.n + 2$  dan diperoleh:

$$Y_{b2} = \begin{bmatrix} y(2) \\ y(5) \\ y(8) \\ y(11) \\ y(14) \end{bmatrix} \text{ dan}$$

$$H_{b2} = \begin{bmatrix} CB(2) & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ CA(5)B(2) & CB(5) & \varepsilon & \varepsilon & \varepsilon \\ CA(8)A(5)B(2) & CA(8)B(5) & CB(8) & \varepsilon & \varepsilon \\ CA(11)A(8)A(5)B(2) & CA(11)A(8)B(5) & CA(11)B(8) & CB(11) & \varepsilon \\ CA(14)A(11)A(8)A(5)B(2) & CA(14)A(11)A(8)B(5) & CA(14)A(11)B(8) & CA(14)B(11) & CB(14) \end{bmatrix}$$

Untuk  $x$  modulo 3 = 2 atau  $s = 2$  menunjukkan bahwa permintaan dipenuhi oleh kapal tanker ke-3 (berkapasitas 22.000 Kl). maka  $x = 3.n + 2$  dengan  $n = 0,1,2,3,4$  sehingga  $k = x + 1 = 3.n + 2 + 1 = 3.n + 3$  dan diperoleh:

$$Y_{b3} = \begin{bmatrix} y(3) \\ y(6) \\ y(9) \\ y(12) \\ y(15) \end{bmatrix} \text{ dan}$$

$$H_{b3} = \begin{bmatrix} CB(3) & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ CA(6)B(3) & CB(6) & \varepsilon & \varepsilon & \varepsilon \\ CA(9)A(6)B(3) & CA(9)B(6) & CB(9) & \varepsilon & \varepsilon \\ CA(12)A(9)A(6)B(3) & CA(12)A(9)B(6) & CA(12)B(9) & CB(12) & \varepsilon \\ CA(15)A(12)A(9)A(6)B(3) & CA(15)A(12)A(9)B(6) & CA(15)A(12)B(9) & CA(15)B(12) & CB(15) \end{bmatrix}$$

Sedangkan untuk volume 22.000 Kl-25.000 Kl jumlah permintaannya sebanyak 11 sehingga  $m_d = 11$ . Jumlah kapal tanker yang bisa digunakan untuk memenuhi permintaan dengan volume 22.000 Kl-25.000 Kl adalah 2 sehingga  $m = 2$ . Urutan permintaan adalah  $x = 0,1,2, \dots, 10$  dengan  $x = 0$  menunjukkan permintaan ke-1 untuk volume 22.000 Kl-25.000 Kl,  $x = 1$  menunjukkan permintaan ke-2 untuk volume 22.000 Kl-25.000 Kl, dan seterusnya sampai  $x = 10$  menunjukkan permintaan ke-11 untuk volume 22.000 Kl-25.000 Kl. Sedangkan  $\gamma$  merupakan hasil bagi  $x$  terhadap 2 dan  $s = 0,1$  merupakan sisa pembagian  $x$  terhadap 2 atau  $x$  modulo 2.

Selanjutnya diperoleh matriks  $Y_c$  dalam urutan permintaan yang sebenarnya ( $k$ ) dan dalam  $x$  sebagai berikut:

$$Y_c(k) \rightarrow Y_c(x)$$

$$\begin{bmatrix} y(1) \\ y(2) \\ y(3) \\ y(4) \\ y(5) \\ y(6) \\ y(7) \\ y(8) \\ y(9) \\ y(10) \\ y(11) \end{bmatrix} \rightarrow \begin{bmatrix} y(0) \\ y(1) \\ y(2) \\ y(3) \\ y(4) \\ y(5) \\ y(6) \\ y(7) \\ y(8) \\ y(9) \\ y(10) \end{bmatrix} = \begin{bmatrix} y(0) \\ y(1) \\ y(0 + 1.2) \\ y(1 + 1.2) \\ y(0 + 2.2) \\ y(1 + 2.2) \\ y(0 + 3.2) \\ y(1 + 3.2) \\ y(0 + 4.2) \\ y(1 + 4.2) \\ y(0 + 5.2) \end{bmatrix}$$

Untuk  $x$  modulo 2 = 0 atau  $s = 0$  menunjukkan bahwa permintaan dipenuhi oleh kapal tanker ke-1 (berkapasitas 25.000 Kl). maka  $x = 2.n$  dengan  $n = 0,1,2,3,4,5$  sehingga  $k = x + 1 = 2.n+1$  dan diperoleh:

$$Y_c 1 = \begin{bmatrix} x(0) \\ x(0 + 2) \\ x(0 + 2.2) \\ x(0 + 3.2) \\ x(0 + 4.2) \\ x(0 + 5.2) \end{bmatrix} = \begin{bmatrix} x(0) \\ x(2) \\ x(4) \\ x(6) \\ x(8) \\ x(10) \end{bmatrix} = \begin{bmatrix} y(1) \\ y(3) \\ y(5) \\ y(7) \\ y(9) \\ y(11) \end{bmatrix} \text{ dan}$$

$$H_c 1 = \begin{bmatrix} CB(1) & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ CA(3)B(1) & CB(3) & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ CA(5)A(3)B(1) & CA(5)B(3) & CB(5) & \varepsilon & \varepsilon & \varepsilon \\ CA(7)A(5)A(3)B(1) & CA(7)A(5)B(3) & CA(7)B(5) & CB(7) & \varepsilon & \varepsilon \\ CA(9)A(7)A(5)A(3)B(1) & CA(9)A(7)A(5)B(3) & CA(9)A(7)B(5) & CA(9)B(7) & CB(9) & \varepsilon \\ CA(11)A(9)A(7)A(5)A(3)B(1) & CA(11)A(9)A(7)A(5)B(3) & CA(11)A(9)A(7)B(5) & CA(11)A(9)B(7) & CA(11)B(9) & CB(11) \end{bmatrix}$$

Untuk  $x$  modulo 2 = 1 atau  $s = 1$  menunjukkan bahwa permintaan dipenuhi oleh kapal tanker ke-2 (berkapasitas 25.000 Kl). maka  $x = 2.n + 1$  dengan  $n = 0,1,2,3,4$  sehingga  $k = x + 1 = 2.n+1+1=2.n+2$  dan diperoleh:

$$Y_c 2 = \begin{bmatrix} x(1) \\ x(1 + 2) \\ x(1 + 2.2) \\ x(1 + 3.2) \\ x(1 + 4.2) \end{bmatrix} = \begin{bmatrix} x(1) \\ x(3) \\ x(5) \\ x(7) \\ x(9) \end{bmatrix} = \begin{bmatrix} y(2) \\ y(4) \\ y(6) \\ y(8) \\ y(10) \end{bmatrix} \text{ dan}$$



$$H_c = \begin{bmatrix} CB(2) & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ CA(4)B(2) & CB(4) & \varepsilon & \varepsilon & \varepsilon \\ CA(6)A(4)B(2) & CA(6)B(4) & CB(6) & \varepsilon & \varepsilon \\ CA(8)A(6)A(4)B(2) & CA(8)A(6)B(4) & CA(8)B(6) & CB(8) & \varepsilon \\ CA(10)A(8)A(6)A(4)B(2) & CA(10)A(8)A(6)B(4) & CA(10)A(8)B(6) & CA(10)B(8) & CB(10) \end{bmatrix}$$

Selanjutnya nilai dari elemen-elemen matriks  $Y_c$  dan  $H_c$  atau yang dinyatakan dengan  $Y_b1$  dan  $H_b1$ ,  $Y_b2$  dan  $H_b2$ ,  $Y_b3$  dan  $H_b3$ ,  $Y_c1$  dan  $H_c1$ , serta  $Y_c1$  dan  $H_c5$  dihitung pada program Scilab dengan menggunakan data-data sekunder yang telah dikumpulkan.

#### 4.7. Data-Data sekunder untuk Model kedua yang digunakan dalam Simulasi Program

Pada model kedua ini data untuk permintaan dengan volume sampai 25.000 K1 diasumsikan berdasarkan data pada model pertama. Ada 26 permintaan dengan tanggal, lama berangkat dan kembali dari pengiriman serta lamanya waktu *loading* dan *unloading* tiap Kilo liter per jam sama seperti pada model pertama namun dari 26 permintaan tersebut diasumsikan hanya 15 permintaan dengan volume kurang dari atau sama dengan 22.000 K1 sedangkan 11 permintaan dengan volume 22.000 K1-25.000 K1. Data asumsi untuk volume permintaan pada model kedua tersebut dapat dilihat pada Lampiran 4.A dengan tanggal dan urutan *customer* sama seperti pada model pertama.

Data-data tersebut selanjutnya disimpan dalam file excel dan diberi nama data\_d. Data yang disimpan dalam file excel dapat dilihat pada Lampiran 4. B. Selanjutnya data-data tersebut digunakan untuk membuat penjadwalan dengan menggunakan program Scilab.

#### 4.8 Penjadwalan Model kedua dengan Program Scilab

Setelah data-data sekunder disimpan dalam file excel selanjutnya dibuat penjadwalan untuk model rantai pasok dengan adanya volume permintaan sampai 25.000 K1 pada program Scilab dengan menggunakan data-data asumsi yang telah disebutkan pada Sub Bab 4.7 dan diberi nama fungsi *supplychain1to2b*. Listing program dari fungsi *supplychain1to2* dapat dilihat pada Lampiran 4.B. Dalam fungsi *supplychain1to2* ini juga menggunakan fungsi-fungsi untuk mengkonversi

data seperti pada model pertama yaitu fungsi `date2intetc`, `int2dateetc` dan `date2stretc`.

Sebagaimana pada model pertama pembuatan jadwal menggunakan program Scilab diawali dengan pembentukan matriks  $Y_c$  dan  $H_c$ . Berdasarkan hasil simulasi dalam program Scilab pada model kedua ini untuk volume permintaan kurang dari atau sama dengan 22.000 Kl dengan  $l = 1,2,3$  diperoleh:

Yb1 =

395100.  
423900.  
446940.  
452700.  
475740.

Yb2 =

400860.  
429660.  
449820.  
464220.  
482940.

Yb3 =

416700.  
435420.  
451260.  
467100.  
485820.



Hb1 =

6600.	- Inf	- Inf	- Inf	- Inf
14520.	6600.	- Inf	- Inf	- Inf
20040.	12120.	4200.	- Inf	- Inf
23760.	15840.	7920.	2400.	- Inf
31680.	23760.	15840.	10320.	6600.

Hb2 =

4200.	- Inf	- Inf	- Inf	- Inf
12120.	6600.	- Inf	- Inf	- Inf
17532.	12012.	4092.	- Inf	- Inf
25104.	19584.	11664.	6600.	- Inf
30036.	24516.	16596.	11532.	3612.

Hb3 =

2832.	- Inf	- Inf	- Inf	- Inf
7176.	3372.	- Inf	- Inf	- Inf
12348.	8544.	4200.	- Inf	- Inf
15828.	12024.	7680.	2160.	- Inf
21348.	17544.	13200.	7680.	4200.

Sedangkan hasil simulasi dalam program Scilab pada model kedua untuk volume permintaan sebesar 22.000 KI-25.000 KI dengan  $l = 1,2$  diperoleh sebagai berikut:

Yc1 =

412380.
425340.
441180.
464220.
478620.
493020.



Yc2 =

423900.

435420.

459900.

475740.

490140.

Hc1 =

6720. - Inf - Inf - Inf - Inf - Inf

6720. 6372. - Inf - Inf - Inf - Inf

14532. 14184. 6840. - Inf - Inf - Inf

22344. 21996. 14652. 6492. - Inf - Inf

30036. 29688. 22344. 14184. 6720. - Inf

30264. 37728. 30384. 22224. 14760. 6720.

Hc2 =

6732. - Inf - Inf - Inf - Inf

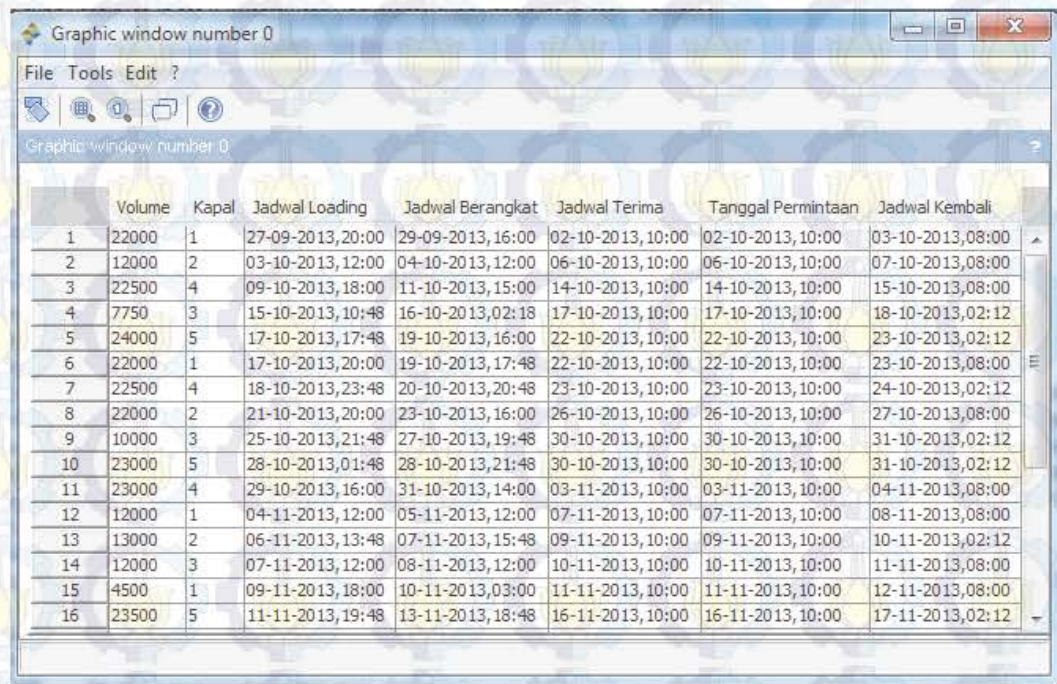
6732. 6492. - Inf - Inf - Inf

14184. 14304. 6612. - Inf - Inf

22116. 22464. 14424. 6732. - Inf

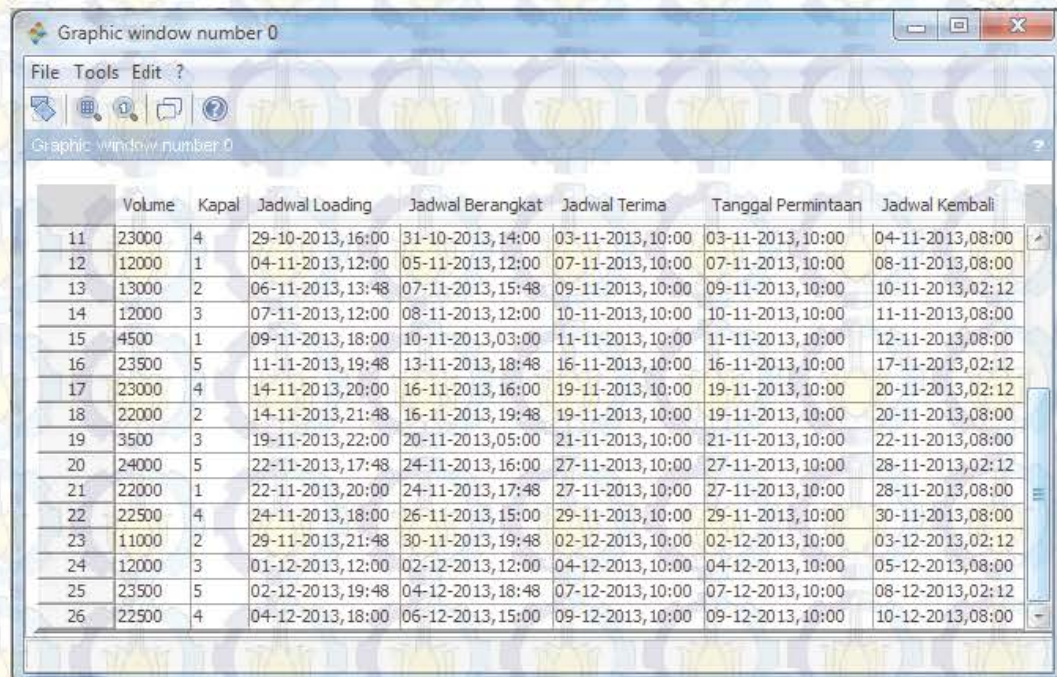
30156. 29928. 21996. 14424. 6612.

Adapun hasil simulasi dari fungsi supplychain1to2b pada program Scilab berupa penjadwalan untuk 15 permintaan dengan volume kurang dari atau sama dengan 22.000 KI dan 11 permintaan dengan volume 22.000 KI-25.000 KI ditunjukkan pada Gambar 4.22 dan Gambar 4.24.



	Volume	Kapal	Jadwal Loading	Jadwal Berangkat	Jadwal Terima	Tanggal Permintaan	Jadwal Kembali
1	22000	1	27-09-2013, 20:00	29-09-2013, 16:00	02-10-2013, 10:00	02-10-2013, 10:00	03-10-2013, 08:00
2	12000	2	03-10-2013, 12:00	04-10-2013, 12:00	06-10-2013, 10:00	06-10-2013, 10:00	07-10-2013, 08:00
3	22500	4	09-10-2013, 18:00	11-10-2013, 15:00	14-10-2013, 10:00	14-10-2013, 10:00	15-10-2013, 08:00
4	7750	3	15-10-2013, 10:48	16-10-2013, 02:18	17-10-2013, 10:00	17-10-2013, 10:00	18-10-2013, 02:12
5	24000	5	17-10-2013, 17:48	19-10-2013, 16:00	22-10-2013, 10:00	22-10-2013, 10:00	23-10-2013, 02:12
6	22000	1	17-10-2013, 20:00	19-10-2013, 17:48	22-10-2013, 10:00	22-10-2013, 10:00	23-10-2013, 08:00
7	22500	4	18-10-2013, 23:48	20-10-2013, 20:48	23-10-2013, 10:00	23-10-2013, 10:00	24-10-2013, 02:12
8	22000	2	21-10-2013, 20:00	23-10-2013, 16:00	26-10-2013, 10:00	26-10-2013, 10:00	27-10-2013, 08:00
9	10000	3	25-10-2013, 21:48	27-10-2013, 19:48	30-10-2013, 10:00	30-10-2013, 10:00	31-10-2013, 02:12
10	23000	5	28-10-2013, 01:48	28-10-2013, 21:48	30-10-2013, 10:00	30-10-2013, 10:00	31-10-2013, 02:12
11	23000	4	29-10-2013, 16:00	31-10-2013, 14:00	03-11-2013, 10:00	03-11-2013, 10:00	04-11-2013, 08:00
12	12000	1	04-11-2013, 12:00	05-11-2013, 12:00	07-11-2013, 10:00	07-11-2013, 10:00	08-11-2013, 08:00
13	13000	2	06-11-2013, 13:48	07-11-2013, 15:48	09-11-2013, 10:00	09-11-2013, 10:00	10-11-2013, 02:12
14	12000	3	07-11-2013, 12:00	08-11-2013, 12:00	10-11-2013, 10:00	10-11-2013, 10:00	11-11-2013, 08:00
15	4500	1	09-11-2013, 18:00	10-11-2013, 03:00	11-11-2013, 10:00	11-11-2013, 10:00	12-11-2013, 08:00
16	23500	5	11-11-2013, 19:48	13-11-2013, 18:48	16-11-2013, 10:00	16-11-2013, 10:00	17-11-2013, 02:12

**Gambar 4.22.** Hasil Simulasi Program berupa Penjadwalan Model kedua untuk Permintaan ke-1 sampai ke-16



	Volume	Kapal	Jadwal Loading	Jadwal Berangkat	Jadwal Terima	Tanggal Permintaan	Jadwal Kembali
11	23000	4	29-10-2013, 16:00	31-10-2013, 14:00	03-11-2013, 10:00	03-11-2013, 10:00	04-11-2013, 08:00
12	12000	1	04-11-2013, 12:00	05-11-2013, 12:00	07-11-2013, 10:00	07-11-2013, 10:00	08-11-2013, 08:00
13	13000	2	06-11-2013, 13:48	07-11-2013, 15:48	09-11-2013, 10:00	09-11-2013, 10:00	10-11-2013, 02:12
14	12000	3	07-11-2013, 12:00	08-11-2013, 12:00	10-11-2013, 10:00	10-11-2013, 10:00	11-11-2013, 08:00
15	4500	1	09-11-2013, 18:00	10-11-2013, 03:00	11-11-2013, 10:00	11-11-2013, 10:00	12-11-2013, 08:00
16	23500	5	11-11-2013, 19:48	13-11-2013, 18:48	16-11-2013, 10:00	16-11-2013, 10:00	17-11-2013, 02:12
17	23000	4	14-11-2013, 20:00	16-11-2013, 16:00	19-11-2013, 10:00	19-11-2013, 10:00	20-11-2013, 02:12
18	22000	2	14-11-2013, 21:48	16-11-2013, 19:48	19-11-2013, 10:00	19-11-2013, 10:00	20-11-2013, 08:00
19	3500	3	19-11-2013, 22:00	20-11-2013, 05:00	21-11-2013, 10:00	21-11-2013, 10:00	22-11-2013, 08:00
20	24000	5	22-11-2013, 17:48	24-11-2013, 16:00	27-11-2013, 10:00	27-11-2013, 10:00	28-11-2013, 02:12
21	22000	1	22-11-2013, 20:00	24-11-2013, 17:48	27-11-2013, 10:00	27-11-2013, 10:00	28-11-2013, 08:00
22	22500	4	24-11-2013, 18:00	26-11-2013, 15:00	29-11-2013, 10:00	29-11-2013, 10:00	30-11-2013, 08:00
23	11000	2	29-11-2013, 21:48	30-11-2013, 19:48	02-12-2013, 10:00	02-12-2013, 10:00	03-12-2013, 02:12
24	12000	3	01-12-2013, 12:00	02-12-2013, 12:00	04-12-2013, 10:00	04-12-2013, 10:00	05-12-2013, 08:00
25	23500	5	02-12-2013, 19:48	04-12-2013, 18:48	07-12-2013, 10:00	07-12-2013, 10:00	08-12-2013, 02:12
26	22500	4	04-12-2013, 18:00	06-12-2013, 15:00	09-12-2013, 10:00	09-12-2013, 10:00	10-12-2013, 08:00

**Gambar 4.23.** Hasil Simulasi Program berupa Penjadwalan Model kedua untuk Permintaan ke-11 sampai ke-26



Gambar 4.21 menunjukkan hasil simulasi dari fungsi *supplychain1to2b* yang berupa jadwal untuk permintaan ke-1 sampai ke-16 untuk permintaan dengan volume sampai 25.000 Kl. Sedangkan Gambar 4.22 menunjukkan hasil simulasi untuk permintaan ke-11 sampai ke-26 untuk permintaan dengan volume sampai 25.000 Kl. Kolom pertama menunjukkan nomer urutan permintaan. Kolom kedua menunjukkan volume permintaan dalam satuan Kl. Kolom ketiga menunjukkan urutan awal kapal tanker yang digunakan untuk pengiriman dengan adanya pengaturan prioritas yaitu pengiriman untuk volume kurang dari atau sama dengan 22.000 Kl diprioritaskan menggunakan kapal tanker dengan kapasitas 22.000 Kl. Kapal tanker 1, 2 dan 3 menunjukkan kapal tanker dengan kapasitas 22.000 Kl. Sedangkan kapal tanker 4 dan 5 menunjukkan kapal tanker dengan kapasitas 25.000 Kl.

Selanjutnya kolom keempat menunjukkan jadwal mulai *loading*. Kolom kelima berisi jadwal keberangkatan kapal tanker. Kolom keenam menunjukkan waktu selesainya proses *unloading* atau diterimanya solar di TBBM *customer*. Kolom ketujuh menunjukkan jam dan tanggal permintaan yang telah ditetapkan oleh TBBM *customer*. Jam dan tanggal pada kolom keenam dan ketujuh sama. Hal ini menunjukkan bahwa jadwal *loading* yang diperoleh dari pengaturan prioritas kapal tanker sesuai kapasitasnya menyebabkan jam dan tanggal selesainya proses *unloading* atau diterimanya solar di TBBM *customer* sesuai dengan permintaan yang telah ditetapkan sebelumnya sehingga tidak terjadi kedatangan lebih awal atau keterlambatan.



## DAFTAR PUSTAKA

Adzkiya, D., (2008), *Membangun Model Petri Net Lampu Lalu Lintas dan Simulasinya*, Tesis Magister Matematika, Institut Teknologi Sepuluh Nopember, Surabaya.

Bagian Proyek Pengembangan Kurikulum Dikmenjur, (2003), *Bahan Bakar*, Direktorat Pendidikan Menengah Kejuruan, Direktorat Jenderal Pendidikan Dasar Dan Menengah, Departemen Pendidikan Nasional, Jakarta. Buku ini dapat juga di download dari Web site  
[http://psbtik.smkn1cms.net/kapal/teknika\\_pelayaran\\_niaga/mengidentifikasi\\_bahan\\_bakar.pdf](http://psbtik.smkn1cms.net/kapal/teknika_pelayaran_niaga/mengidentifikasi_bahan_bakar.pdf)

Elmahi, I., Grunder, O., dan Elmoudni, A., (2003), “A *Max Plus Algebra Approach for Modelling and Control a Supply Chain*”, *Proceedings of 2003, IEEE Conference on (volume 2)*, hal. 1425-1430.

Irawan, A. P., (2008), *Manajemen Rantai pasok*, Buku Ajar Manajemen Rantai Pasok, Universitas Tarumanegara, Jakarta. Buku ini dapat juga di download dari Web site  
[http://api2012.weebly.com/uploads/1/2/3/1/12314186/agustinus\\_purna\\_irawan\\_diktat\\_manajemen\\_rantai\\_pasokan\\_2008.pdf](http://api2012.weebly.com/uploads/1/2/3/1/12314186/agustinus_purna_irawan_diktat_manajemen_rantai_pasokan_2008.pdf)

Pertamina, (2012), *Supply & Distribution Region III*, Pertamina, Surabaya.

Laporan Tahunan Pertamina, (2012), *Entrusted with The New Energy*, Pertamina, Jakarta. Buku ini dapat juga di download dari Web site  
<http://www.pertamina.com/media/84ae4174-db0d-4b4c-ada4-628e82f5606d/AR%2019%20mar%202013.pdf>

Sierliawati, W. P., (2014), *Rancangan dan Analisis Distribusi Pasokan Bahan Bakar Minyak Menggunakan Pendekatan Petri net dan Aljabar Max Plus*, Tesis Magister Matematika, Institut Teknologi Sepuluh Nopember, Surabaya.

Subiono, (2014), *Aljabar Min Max-plus dan Terapannya*, Buku Ajar Aljabar Max-Plus, Institut Teknologi Sepuluh Nopember, Surabaya.

## BAB 5

### KESIMPULAN

Pada bab ini diberikan kesimpulan dari hasil penelitian yang dilakukan yaitu tentang pemodelan rantai pasok distribusi solar dari TBBM Tuban sebagai *supplier* menuju TBBM Manggis di Bali dan Tanjung Wangi di Banyuwangi sebagai TBBM-TBBM *customer* menggunakan *Petri Net* dan Aljabar *Max Plus* dengan mempertimbangkan prioritas kapal tanker serta penjadwalan *loading* dan keberangkatan kapal tanker menggunakan program Scilab. Pada bab ini juga diberikan saran untuk penelitian selanjutnya yang merupakan pengembangan dari penelitian ini.

#### 5.1 Kesimpulan

Berdasarkan rumusan masalah, perhitungan serta analisis yang dilakukan pada penelitian ini maka dapat disimpulkan hasil sebagai berikut:

1. Model *Petri Net* pertama untuk kapasitas kapal tanker yang diasumsikan sama dibuat dengan 14 *place* dan 12 transisi. Sedangkan model *Petri Net* kedua untuk perbedaan kapasitas kapal tanker dipertimbangkan dibuat dengan 24 *place* dan 24 transisi.
2. Berdasarkan model *Petri Net* yang diperoleh maka diperoleh penjabaran dalam model Aljabar *Max Plus* sebagai berikut:

Model pertama:

$$t_m(k) = \begin{cases} t_n(k-n) \otimes p_d(k-n) \oplus u(k), & \text{untuk } k > n \\ u(k), & \text{untuk } 1 \leq k \leq n \end{cases}$$

$$t_n(k) = t_m(k) \otimes p_a(k) \otimes p_b(k) \otimes p_c(k)$$

$$y(k) = t_n(k):$$

Model kedua:

$$t_{xc}(k) = \begin{cases} t_{yc}(k-m) \otimes p_{ic}(k-m) \oplus u_c(k), & \text{untuk } k > m \\ u_c(k), & \text{untuk } 1 \leq k \leq m \end{cases}$$

$$t_{yc}(k) = t_{xc}(k) \otimes p_{fc}(k) \otimes p_{gc}(k) \otimes p_{hc}(k)$$

$$y_c(k) = t_{yc}(k)$$

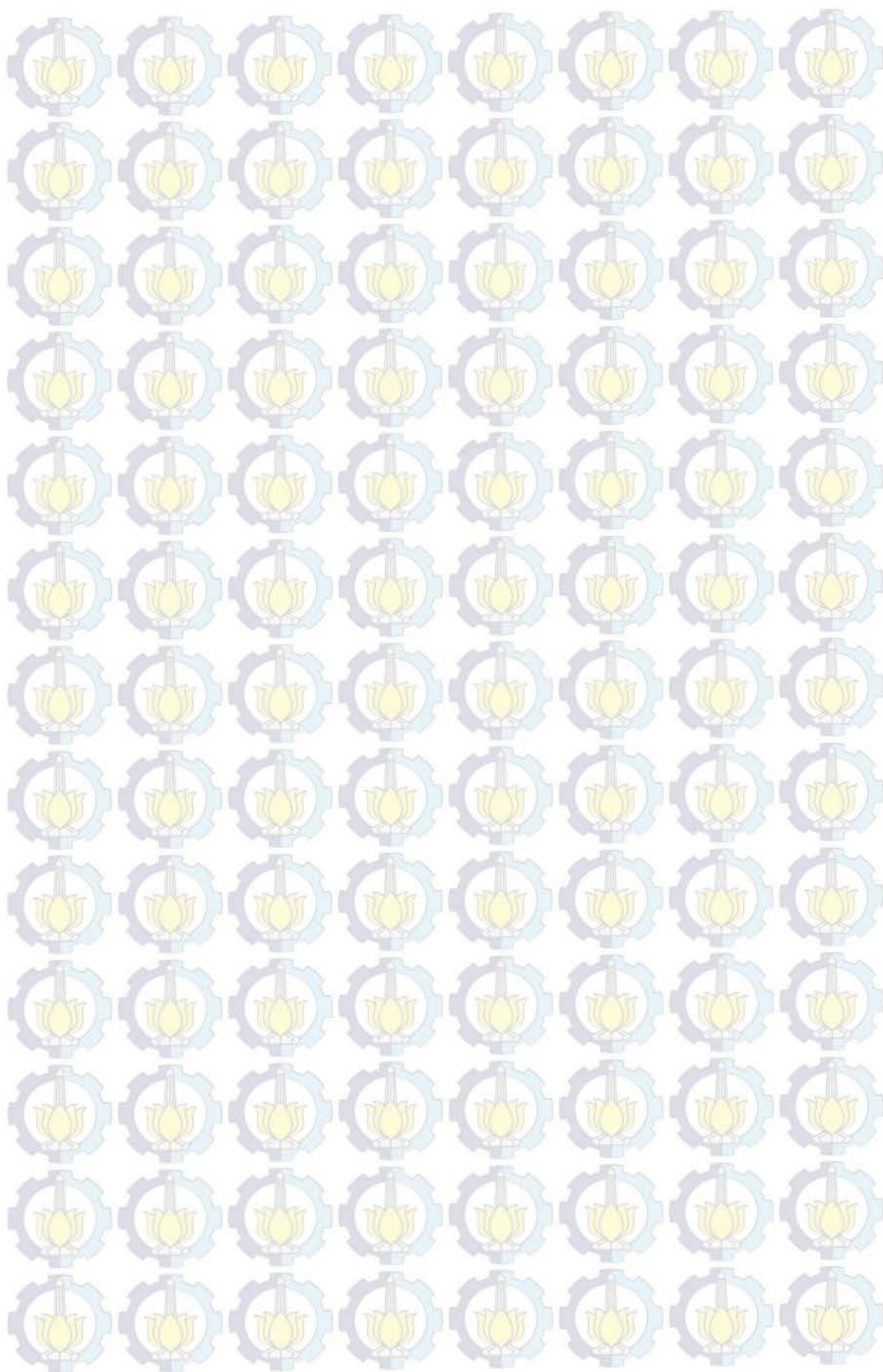


3. Ada tiga fungsi yang dibuat untuk mengkonversi data yaitu fungsi *date2intetc* untuk mengkonversi data berupa menit, jam dan tanggal ke bentuk integer, fungsi *int2dateetc* untuk mengkonversi dari bentuk integer ke bentuk menit, jam dan tanggal, serta fungsi *date2stretc* untuk mengkonversi dari bentuk menit, jam dan tanggal ke bentuk string.
4. Pada model pertama fungsi untuk membuat jadwal *loading* dan keberangkatan kapal tanker menggunakan program Scilab diberi nama *supplychain1to2*. Sedangkan fungsi pada model kedua diberi nama *supplychain1to2b*. Hasil simulasi dari fungsi *supplychain1to2* dan *supplychain1to2b* berupa jadwal *loading*, keberangkatan kapal tanker, sampainya kapal tanker di TBBM *customer* dan di TBBM *supplier* setelah kembali dari pengiriman solar ke TBBM *customer*. Dengan menggunakan jadwal *loading* dan keberangkatan kapal tanker yang diperoleh menunjukkan bahwa waktu diterimanya solar di TBBM *customer* sesuai dengan waktu yang telah diminta sebelumnya.
5. Pada model pertama dengan memprioritaskan pengiriman menggunakan kapal tanker yang telah kembali dari pengiriman sebelumnya maka 26 permintaan dapat dipenuhi tepat waktu hanya menggunakan 4 kapal tanker. Sedangkan pada model kedua dengan memprioritaskan pengiriman untuk volume tidak melebihi 22.000 Kl menggunakan kapal tanker berkapasitas 22.000 Kl maka 26 permintaan dapat dipenuhi tepat waktu dengan menggunakan 5 kapal tanker.

## 5.2 Saran

Pada penelitian ini dibuat model *Petri Net* dan Aljabar *Max Plus* dari 1 *supplier* menuju 2 *customer* dengan kapasitas dari 5 kapal tanker dianggap sama. Hal ini dikarenakan data-data sekunder yang diperoleh menunjukkan tidak ada volume permintaan yang melebihi 22.000 Kl. Selanjutnya dibuat model *Petri Net* dengan mempertimbangkan perbedaan kapasitas kapal tanker yaitu 2 kapal tanker berkapasitas 25.000 Kl dan sedangkan 3 lainnya 22.000 Kl. Pada penelitian ini diasumsikan tidak ada gangguan perjalanan kapal tanker.

Oleh karena itu, pada penelitian berikutnya dapat dibuat rancangan penjadwalan dengan data baru menggunakan model *Petri Net* dan Aljabar *Max Plus* dengan mempertimbangkan kapasitas kapal tanker, atau dapat juga dibuat model rantai pasok dan rancangan penjadwalan untuk kemungkinan adanya volume permintaan melebihi kapasitas kapal tanker atau model rantai pasok dengan *supplier* dan *customer* yang lebih banyak. Selain itu, dapat dibuat model rantai pasok dengan kemungkinan adanya gangguan perjalanan kapal tanker.





# Lampiran 1.A

## Data Volume dan Waktu Permintaan untuk Model pertama

No	TBBM	Volume	Tanggal	Jam
1	Manggis	22000	02-Okt-13	10:00
2	Manggis	12000	06-Okt-13	10:00
3	Manggis	12000	14-Okt-13	10:00
4	Tanjungwangi	7723	17-Okt-13	10:00
5	Tanjungwangi	17500	22-Okt-13	10:00
6	Manggis	22000	22-Okt-13	10:00
7	Tanjungwangi	5000	23-Okt-13	10:00
8	Manggis	22000	26-Okt-13	10:00
9	Tanjungwangi	10000	30-Okt-13	10:00
10	Manggis	12000	30-Okt-13	10:00
11	Tanjungwangi	6000	03-Nop-13	10:00
12	Manggis	12000	07-Nop-13	10:00
13	Tanjungwangi	13000	09-Nop-13	10:00
14	Manggis	12000	10-Nop-13	10:00
15	Manggis	4500	11-Nop-13	10:00
16	Tanjungwangi	5000	16-Nop-13	10:00
17	Tanjungwangi	14000	19-Nop-13	10:00
18	Manggis	22000	19-Nop-13	10:00
19	Manggis	3500	21-Nop-13	10:00
20	Tanjungwangi	14000	27-Nop-13	10:00
21	Manggis	22000	27-Nop-13	10:00
22	Manggis	3500	29-Nop-13	10:00
23	Tanjungwangi	11000	02-Des-13	10:00
24	Manggis	12000	04-Des-13	10:00
25	Tanjungwangi	6000	07-Des-13	10:00
26	Manggis	11500	09-Des-13	10:00

Lampiran 1.B

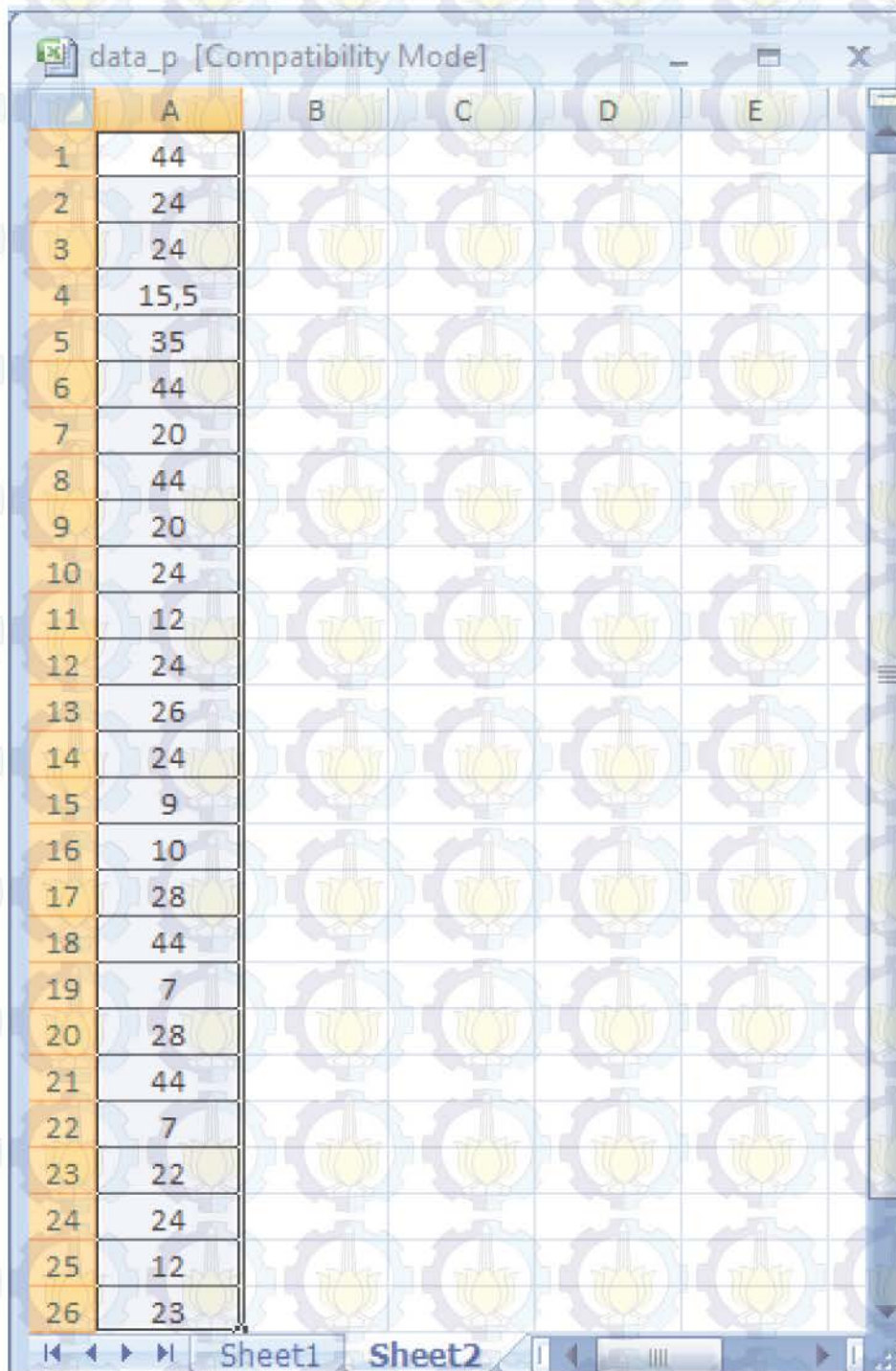
Data yang disimpan dalam File Excel untuk Model pertama

Sheet 1: Waktu Permintaan

	A	B	C	D	E
1	Minutes	Hours	Day	Month	Year
2	0	10	2	10	2013
3	0	10	6	10	2013
4	0	10	14	10	2013
5	0	10	17	10	2013
6	0	10	22	10	2013
7	0	10	22	10	2013
8	0	10	23	10	2013
9	0	10	26	10	2013
10	0	10	30	10	2013
11	0	10	30	10	2013
12	0	10	3	11	2013
13	0	10	7	11	2013
14	0	10	9	11	2013
15	0	10	10	11	2013
16	0	10	11	11	2013
17	0	10	16	11	2013
18	0	10	19	11	2013
19	0	10	19	11	2013
20	0	10	21	11	2013
21	0	10	27	11	2013
22	0	10	27	11	2013
23	0	10	29	11	2013
24	0	10	2	12	2013
25	0	10	4	12	2013
26	0	10	7	12	2013
27	0	10	9	12	2013



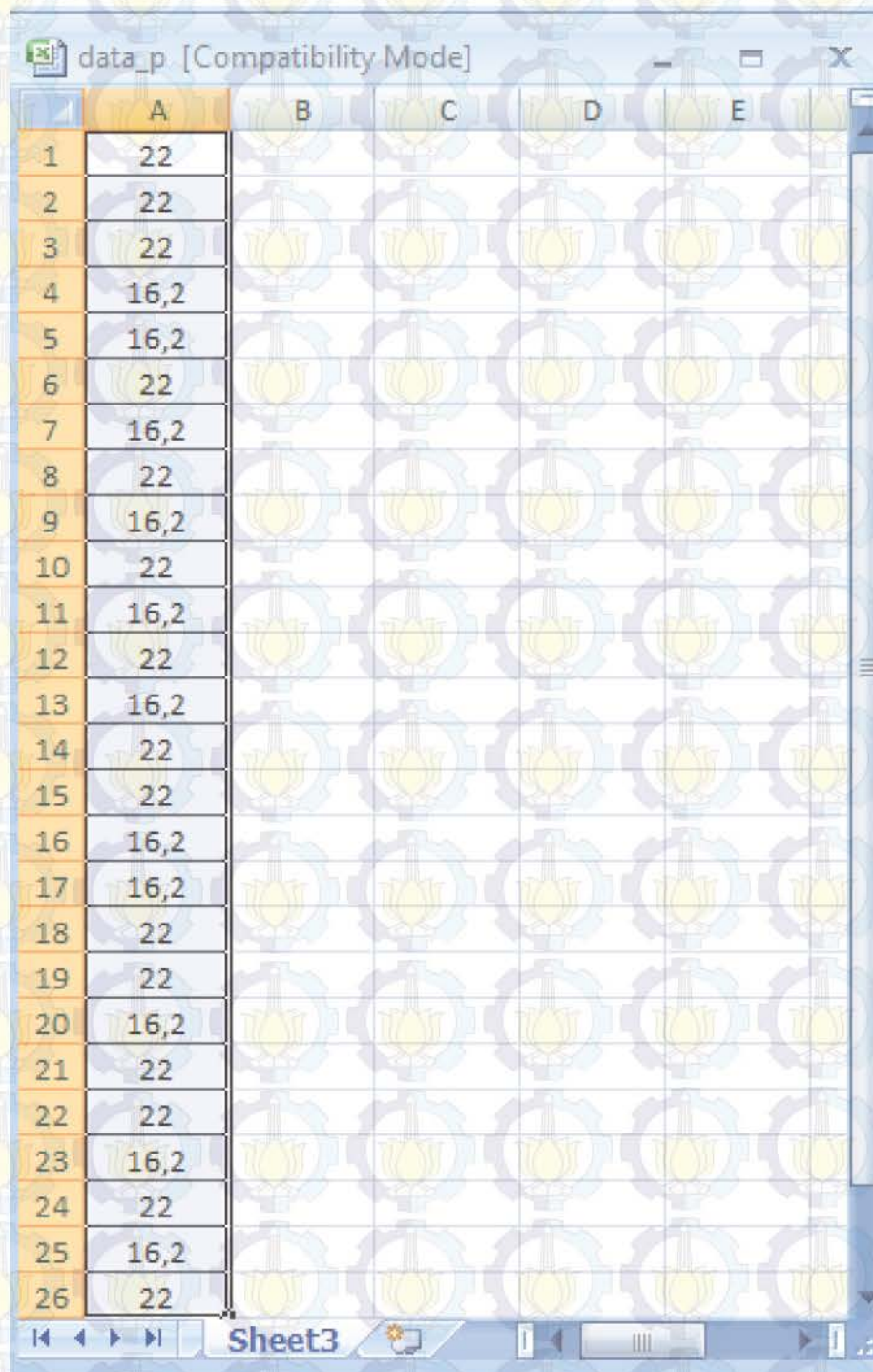
Sheet 2: Lama Loading atau Unloading



	A	B	C	D	E
1	44				
2	24				
3	24				
4	15,5				
5	35				
6	44				
7	20				
8	44				
9	20				
10	24				
11	12				
12	24				
13	26				
14	24				
15	9				
16	10				
17	28				
18	44				
19	7				
20	28				
21	44				
22	7				
23	22				
24	24				
25	12				
26	23				



Sheet 3: Lama Perjalanan Berangkat dan Pulang dari TBBM *Customer*



	A	B	C	D	E	F
1	22					
2	22					
3	22					
4	16,2					
5	16,2					
6	22					
7	16,2					
8	22					
9	16,2					
10	22					
11	16,2					
12	22					
13	16,2					
14	22					
15	22					
16	16,2					
17	16,2					
18	22					
19	22					
20	16,2					
21	22					
22	22					
23	16,2					
24	22					
25	16,2					
26	22					

## LAMPIRAN 2.A

### Listing Program Fungsi date2intetc

```
function x=date2intetc(mi,h,d,m,y,start)
M=[31,28,31,30,31,30,31,31,30,31,30,31];
Y=365;
x=0;
for i=start:1:y-1
    if modulo(i,100)==0 then
        if modulo(i,400)==0 then
            numday=366;
        else
            numday=365;
        end
    elseif modulo(i,4)==0 then
        numday=366;
    else
        numday=365;
    end
    x=x+numday;
end
if modulo(y,100)==0 then
    if modulo(y,400)==0 then
        M(2)=29;;
    else
        M(2)=28;
    end
elseif modulo(y,4)==0 then
    M(2)=29;
else
    M(2)=28;
end
for i=1:m-1
    x=x+M(i);
end
x=x+d;
x=(x-1)*1440+(h-1)*60+(mi);
endfunction
```



## LAMPIRAN 2.B

### Listing Program Fungsi int2dateetc

```
function [mi, h, d, m, y]=int2dateetc(z, start)
M=[31,28,31,30,31,30,31,31,30,31,30,31];
Y=365;
y=start;
m=1;
d=1;
x=ceil(z/1440);
while 1
    if modulo(y,100)==0 then
        if modulo(y,400)==0 then
            numday=366;
        else
            numday=365;
        end
    elseif modulo(y,4)==0 then
        numday=366;
    else
        numday=365;
    end
    if x<=numday then
        break;
    end
    x=x-numday;
    y=y+1;
end
if modulo(y,100)==0 then
    if modulo(y,400)==0 then
        M(2)=29;;
    else
        M(2)=28;
    end
elseif modulo(y,4)==0 then
    M(2)=29;
else
    M(2)=28;
end
for i=1:12
    if x<=M(i) then
        m=i;
        d=x;
        break;
    end
    x=x-M(i);
end
if z<=60 then
    mi=z;
else
    mi=z-(60*fix(z/60));
end
if z<=1440 then
    h=1+fix(z/60);
else
    h=1+fix(modulo(z,1440)/60);
end
endfunction
```



### LAMPIRAN 3.A

#### Listing Program Fungsi date2stretc

```
function s=date2stretc(d, m, y, h, mi);  
    d=string(d);  
    m=string(m);  
    y=string(y);  
    h=string(h);  
    mi=string(mi);  
  
    if length(d)==1 then  
        d='0'+d;  
    end  
    if length(m)==1 then  
        m='0'+m;  
    end  
    if length(h)==1 then  
        h='0'+h;  
    end  
    if length(mi)==1 then  
        mi='0'+mi;  
    end  
  
    s=d+'-'+m+'-'+y+' '+h+'-'+mi;  
endfunction
```

### LAMPIRAN 3.B

#### Listing Program Fungsi supplychain1to2

```
function supplychain1to2
//Mendefinisikan n, nd dan e
n=5;
nd=26;
e=-%inf;

//Input data dari file xls
Sheets = readxls("data_p.xls");

//Data tanggal permintaan
s1= Sheets(1);
tanggal_permintaan=s1.value;

//Merubah data tanggal, bulan dan tahun ke bentuk integer
data_Y=zeros(nd,1);
for i=1:nd
    data_Y(i,1)=date2intc(tanggal_permintaan(i+1,1),tanggal_permintaan(i+1,2),tanggal_permintaan(i+1,3),tanggal_permintaan(i+1,4),tanggal_permintaan(i+1,5),2013);
end

//Data lama proses loading atau unloading
s2= Sheets(2);
lo=s2.value;
load_unload=lo*60;

//Data lama berangkat atau kembali
s3 = Sheets(3);
be=s3.value;
berangkat_kembali=be*60;

p=load_unload+berangkat_kembali+load_unload;
pd = berangkat_kembali;

//Mendefinisikan matrik A, B dan C
A = zeros(2,2*nd)
for i = 6:nd
    A(:,2*i-1:2*i)=[e pd(i-n);e maxplusotimes(p(i),pd(i-n))];
end

B = zeros(2,nd)
for i = 1:nd
    B(:,i)=[e;p(i)];
end

C=[e 0];

//Membentuk matriks Y
Y = zeros(n+1,n);
for i = 0:n
    Y(i+1,1) = data_Y(1+n*i);
end

for j = 2:n
    for i = 0:n-1
        Y(i+1,j) = data_Y(1+(n*i+(j-1)));
    end
end
```



```

//Menampilkan Y
//for i=1:n
//if i==1 then
//disp("Y"+string(i)+"=")
//disp(Y(:,i))
//else
//disp("Y"+string(i)+"=")
//disp(Y(1:n-1,i))
//end
//end

//Membentuk matriks H1
H1 = maxpluszeros(n+1,n+1);
for i = 1:n+1
    for j=1:n+1
        if i==j then
            H1(i,j)=maxplusotimes(C,B(:,1+(i-1)*n));

        elseif i==j+1 then
            H1(i,j)=maxplusotimes(C, maxplusotimes(A(:,10*i-9:10*i-8),B(:,1+(i-2)*n)));

        elseif i == j+2 then
            H1(i,j) = maxplusotimes(maxplusotimes(C, A(:,10*i-9:10*i-8)), maxplusotimes(A(:,10*i-19:10*i-18),B(:,1+(i-3)*n)))

        elseif i == j+3 then
            H1(i,j) = maxplusotimes(C, maxplusotimes(maxplusotimes(A(:,10*i-9:10*i-8),A(:,10*i-19:10*i-18)), maxplusotimes(A(:,10*i-29:10*i-28),B(:,1+(i-4)*n))))

        elseif i == j+4 then
            H1(i,j) = maxplusotimes(maxplusotimes(C,A(:,10*i-9:10*i-8)), maxplusotimes(maxplusotimes(A(:,10*i-19:10*i-18),A(:,10*i-29:10*i-28)), maxplusotimes(A(:,10*i-39:10*i-38),B(:,1+(i-5)*n))))

        elseif i == j+5 then
            H1(i,j) = maxplusotimes(maxplusotimes(C,maxplusotimes(A(:,10*i-9:10*i-8),A(:,10*i-19:10*i-18))), maxplusotimes(maxplusotimes(A(:,10*i-29:10*i-28),A(:,10*i-39:10*i-38)), maxplusotimes(A(:,10*i-49:10*i-48),B(:,1+(i-6)*n))))

        else
            end
        end
    end
end

//Menampilkan H1
//disp("H1");
//disp(H1);

//Menentukan waktu mulai loading
//Menghitung nilai U1
H2 = maxpluszeros(n,n);
U = zeros(n+1,n);
U(:,1)=maxplusinsol(H1,Y(:,1));
for k = 2:n
    for i = 1:n
        for j=1:n
            if i==j then
                H2(i,j)=maxplusotimes(C,B(:,k+(i-1)*n));

            elseif i==j+1 then
                H2(i,j)=maxplusotimes(C, maxplusotimes(A(:,10*i-(-2*k+11):10*i-(-2*k+10)),B(:,k+(i-2)*n)));

```



```

elseif i == j+2 then
    H2(i,j) = maxplusotimes(maxplusotimes(C, A(:,10*i-(-2*k+11):10*i-(-2*k+10))),
maxplusotimes(A(:,10*i-(-2*k+21):10*i-(-2*k+20)),B(:,k+(i-3)*n)))

elseif i == j+3 then
    H2(i,j) = maxplusotimes(C, maxplusotimes(maxplusotimes(A(:,10*i-(-2*k+11):10*i-(-2*k+10)),A(:,10*i-(-2*k+21):10*i-(-2*k+20))), maxplusotimes(A(:,10*i-(-2*k+31):10*i-(-2*k+30)),B(:,k+(i-4)*n))))

elseif i == j+4 then
    H2(i,j) = maxplusotimes(maxplusotimes(C,A(:,10*i-(-2*k+11):10*i-(-2*k+10))),
maxplusotimes(maxplusotimes(A(:,10*i-(-2*k+21):10*i-(-2*k+20)),A(:,10*i-(-2*k+31):10*i-(-2*k+30))),
maxplusotimes(A(:,10*i-(-2*k+41):10*i-(-2*k+40)),B(:,k+(i-5)*n))))

else
end
end
end
end
//Menampilkan H2,H3,H4 dan H5
//disp("H"+string(k)+"=");
//disp(H2);
U(1:n,k)=maxplusinsol(H2,Y(1:n,k));

end
//Menampilkan U1,U2,U3,U4,dan U5
//for i=1:n
//if i==1 then
//disp("U"+string(i)+"=")
//disp(U(:,i))
//else
//disp("U"+string(i)+"=")
//disp(U(1:n-1,i))
//end
//end

//Menentukan waktu loading
Utemp = matrix(U',1,nd+n-1);
Uakhir = Utemp(1:nd);

//Menentukan waktu berangkat
berangkat=Uakhir+load_unload;

//Menentukan waktu produk diterima
terima=berangkat+berangkat_kembali+load_unload;

//Menentukan waktu kembali dari pengiriman
pulang=terima+berangkat_kembali

//Merubah waktu loading, berangkat, produk diterima dan kembali dari pengiriman ke bentuk menit, jam,
tanggal, bulan dan tahun
tanggal_load=zeros(nd,5);
for i= 1:nd
[tanggal_load(i,1),tanggal_load(i,2),tanggal_load(i,3),tanggal_load(i,4),tanggal_load(i,5)]=int2dateetc(Uakhir
(i,1),2013);
end

tanggal_berangkat=zeros(nd,5);
for i= 1:nd

```

```

[tanggal_berangkat(i,1),tanggal_berangkat(i,2),tanggal_berangkat(i,3),tanggal_berangkat(i,4),tanggal_berang
kat(i,5)]=int2dateetc(berangkat(i,1),2013);
end

tanggal_terima=zeros(nd,5);
for i= 1:nd

[tanggal_terima(i,1),tanggal_terima(i,2),tanggal_terima(i,3),tanggal_terima(i,4),tanggal_terima(i,5)]=int2date
etc(terima(i,1),2013);
end

tanggal_pulang=zeros(nd,5);
for i= 1:nd

[tanggal_pulang(i,1),tanggal_pulang(i,2),tanggal_pulang(i,3),tanggal_pulang(i,4),tanggal_pulang(i,5)]=int2da
teetc(pulang(i,1),2013);
end

//Menentukan Selisih tanggal permintaan dengan sampainya produk di TBBM customer
selisih=data_Y-terima

//Menentukan urutan kapal awal
kapal=zeros(nd,1);
for i=1:n
    for j=0:n
        kapal(i+j*5,1)=i
    end
end
kapal=kapal(1:nd,1);

//Menentukan urutan kapal yang baru
urutan_kapal=zeros(nd,1);
for i=1
    urutan_kapal(i,1)=1
end

for i=2:nd
    if pulang(i-1,1)<= Uakhir(i,1) then
        urutan_kapal(i,1)=1
    elseif urutan_kapal(i-1,1)==1 then
        urutan_kapal(i,1)=2
    elseif urutan_kapal(i-1,1)==2 then
        if pulang(i-2,1)<= Uakhir(i,1) then
            urutan_kapal(i,1)=1
        else
            urutan_kapal(i,1)=3
        end
    elseif urutan_kapal(i-1,1)==3 then
        if pulang(i-3,1)<= Uakhir(i,1) then
            urutan_kapal(i,1)=urutan_kapal(i-3,1)
        elseif pulang(i-2,1)<= Uakhir(i,1) then
            urutan_kapal(i,1)=urutan_kapal(i-2,1)
        else
            urutan_kapal(i,1)=4
        end
    elseif urutan_kapal(i-1,1)==4 then
        if pulang(i-4,1)<= Uakhir(i,1) then
            urutan_kapal(i,1)=urutan_kapal(i-4,1)
        elseif pulang(i-3,1)<= Uakhir(i,1) then
            urutan_kapal(i,1)=urutan_kapal(i-3,1)
        elseif pulang(i-2,1)<= Uakhir(i,1) then
            urutan_kapal(i,1)=urutan_kapal(i-2,1)
        end
    end
end

```



```

        else
            urutan_kapal(i,1)=5
        end
    else
        end
    end
end

//Menampilkan hasil di tabel
number=[1:nd];
number=string(number);
kapal=string(kapal);
selisih=string(selisih);
urutan_kapal=string(urutan_kapal);
jadwal_load=[];
jadwal_berangkat=[];
jadwal_terima=[];
jadwal_permintaan=[];
jadwal_kembali=[];
for i=1:nd

    jadwal_load=[jadwal_load;date2stretc(tanggal_load(i,3),tanggal_load(i,4),tanggal_load(i,5),tanggal_load(i,2),
    tanggal_load(i,1))];

    jadwal_berangkat=[jadwal_berangkat;date2stretc(tanggal_berangkat(i,3),tanggal_berangkat(i,4),tanggal_bera
    ngkat(i,5),tanggal_berangkat(i,2),tanggal_berangkat(i,1))];

    jadwal_terima=[jadwal_terima;date2stretc(tanggal_terima(i,3),tanggal_terima(i,4),tanggal_terima(i,5),tanggal
    _terima(i,2),tanggal_terima(i,1))];

    jadwal_permintaan=[jadwal_permintaan;date2stretc(tanggal_permintaan(i+1,3),tanggal_permintaan(i+1,4),ta
    nggal_permintaan(i+1,5),tanggal_permintaan(i+1,2),tanggal_permintaan(i+1,1))];

    jadwal_kembali=[jadwal_kembali;date2stretc(tanggal_pulang(i,3),tanggal_pulang(i,4),tanggal_pulang(i,5),ta
    nggal_pulang(i,2),tanggal_pulang(i,1))];
end

    data=[['No.','Jadwal Loading','Urutan Awal Kapal','Jadwal Keberangkatan','Jadwal Tiba','Tanggal
    Permintaan','Selisih','Jadwal Kembali','Urutan Akhir
    Kapal'];[number,jadwal_load,kapal,jadwal_berangkat,jadwal_terima,jadwal_permintaan,selisih,jadwal_kemb
    ali,urutan_kapal]];

    f = gcf();
    f.position=[10,10,900,400]
    clf
    as = f.axes_size; // [width height]
    ut = uicontrol("style","table",...
        "string",data,...
        "position",[5 as(2)-300 800 287],... // => @top left corner of figure
        "tooltipstring","Data from majors towns")
    delete(ut);
    ut = uicontrol("style","table",...
        "string",data,...
        "position",[5 as(2)-300 800 287],... // => @top left corner of figure
        "tooltipstring","Data from majors towns")
endfunction

```




#### Lampiran 4.A

##### Data Asumsi Volume dan Waktu permintaan untuk model kedua

No	TBBM	Volume	Tanggal Permintaan	Jam
1	Manggis	22000	02-Okt-13	10:00
2	Manggis	12000	06-Okt-13	10:00
3	Manggis	12000	14-Okt-13	10:00
4	Tanjung Wangi	7723	17-Okt-13	10:00
5	Tanjung Wangi	17500	22-Okt-13	10:00
6	Manggis	22000	22-Okt-13	10:00
7	Tanjung Wangi	5000	23-Okt-13	10:00
8	Manggis	22000	26-Okt-13	10:00
9	Tanjung Wangi	10000	30-Okt-13	10:00
10	Manggis	12000	30-Okt-13	10:00
11	Tanjung Wangi	6000	03-Nop-13	10:00
12	Manggis	12000	07-Nop-13	10:00
13	Tanjung Wangi	13000	09-Nop-13	10:00
14	Manggis	12000	10-Nop-13	10:00
15	Manggis	4500	11-Nop-13	10:00
16	Tanjung Wangi	5000	16-Nop-13	10:00
17	Tanjung Wangi	14000	19-Nop-13	10:00
18	Manggis	22000	19-Nop-13	10:00
19	Manggis	3500	21-Nop-13	10:00
20	Tanjung Wangi	14000	27-Nop-13	10:00
21	Manggis	22000	27-Nop-13	10:00
22	Manggis	3500	29-Nop-13	10:00
23	Tanjung Wangi	11000	02-Des-13	10:00
24	Manggis	12000	04-Des-13	10:00
25	Tanjung Wangi	6000	07-Des-13	10:00
26	Manggis	11500	09-Des-13	10:00

Keterangan warna:

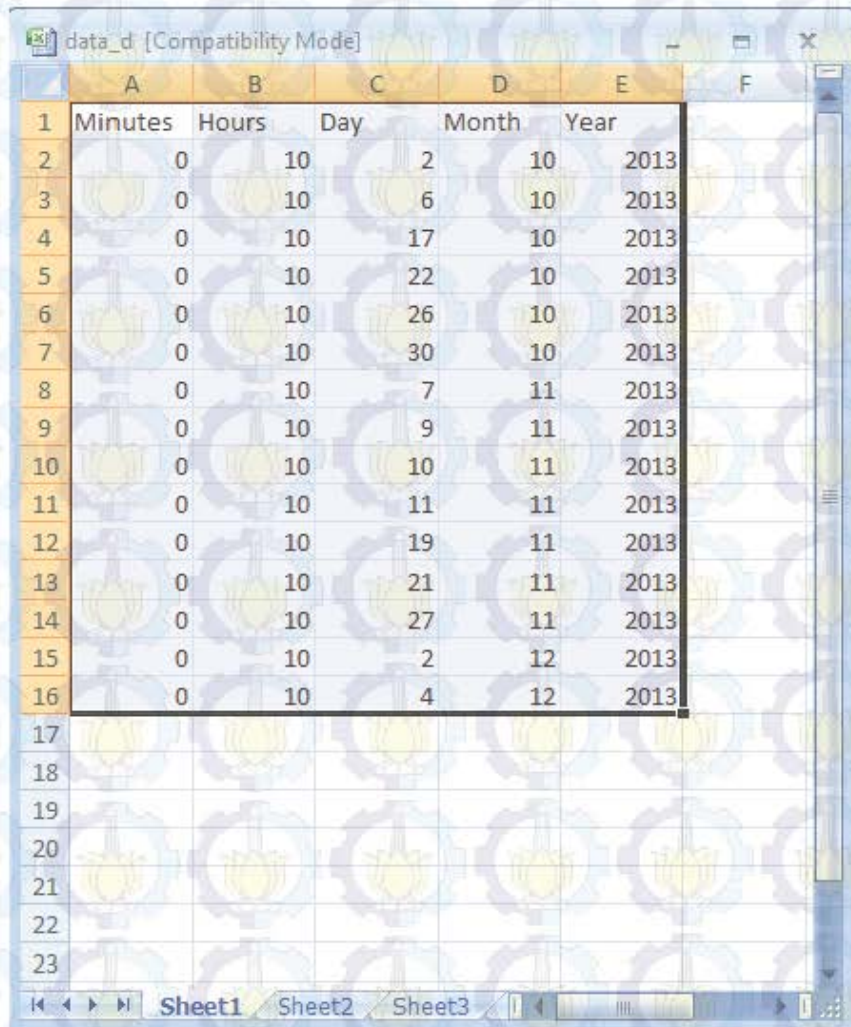
 : permintaan dengan volume kurang dari atau sama dengan 22.000 Kl

 : permintaan dengan volume 22.000 Kl-25.000 Kl

#### Lampiran 4.B

Data yang disimpan dalam File Excel untuk Model kedua

Sheet 1: Waktu Permintaan untuk volume  $\leq 22.000$  K1



	A	B	C	D	E	F
1	Minutes	Hours	Day	Month	Year	
2	0	10	2	10	2013	
3	0	10	6	10	2013	
4	0	10	17	10	2013	
5	0	10	22	10	2013	
6	0	10	26	10	2013	
7	0	10	30	10	2013	
8	0	10	7	11	2013	
9	0	10	9	11	2013	
10	0	10	10	11	2013	
11	0	10	11	11	2013	
12	0	10	19	11	2013	
13	0	10	21	11	2013	
14	0	10	27	11	2013	
15	0	10	2	12	2013	
16	0	10	4	12	2013	
17						
18						
19						
20						
21						
22						
23						

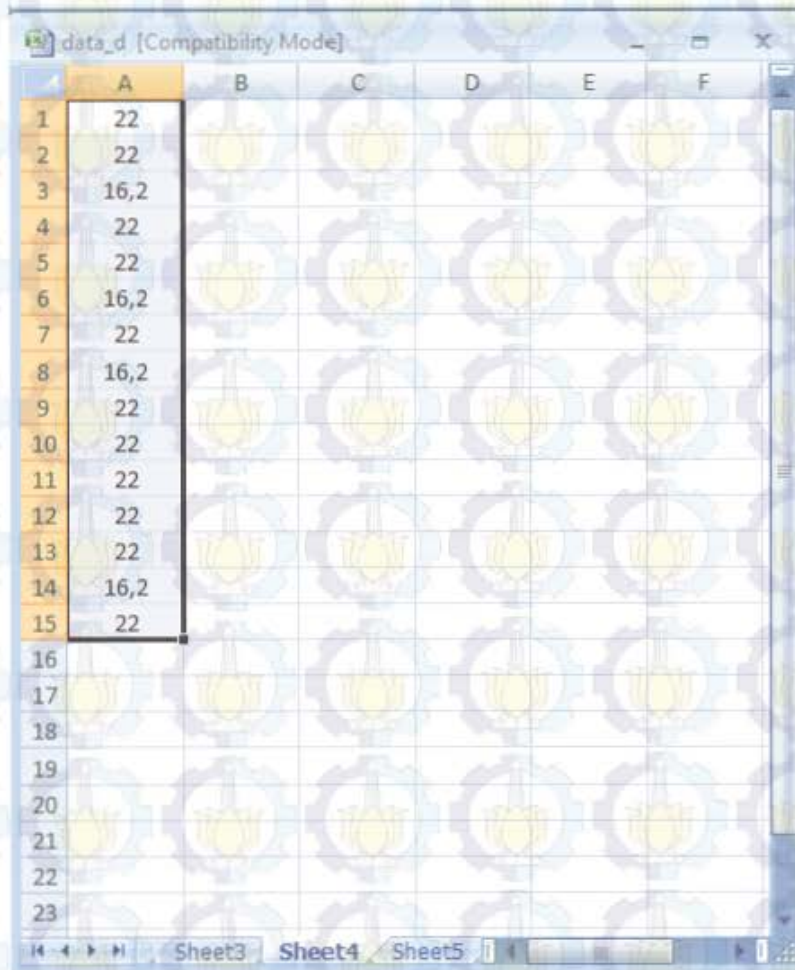


Sheet 3: Lama *Loading* atau *Unloading* untuk Volume Permintaan  $\leq 22.000$  KI

	A	B	C	D	E	F
1	44					
2	24					
3	15,5					
4	44					
5	44					
6	20					
7	24					
8	26					
9	24					
10	9					
11	44					
12	7					
13	44					
14	22					
15	24					
16						
17						
18						
19						
20						
21						
22						
23						

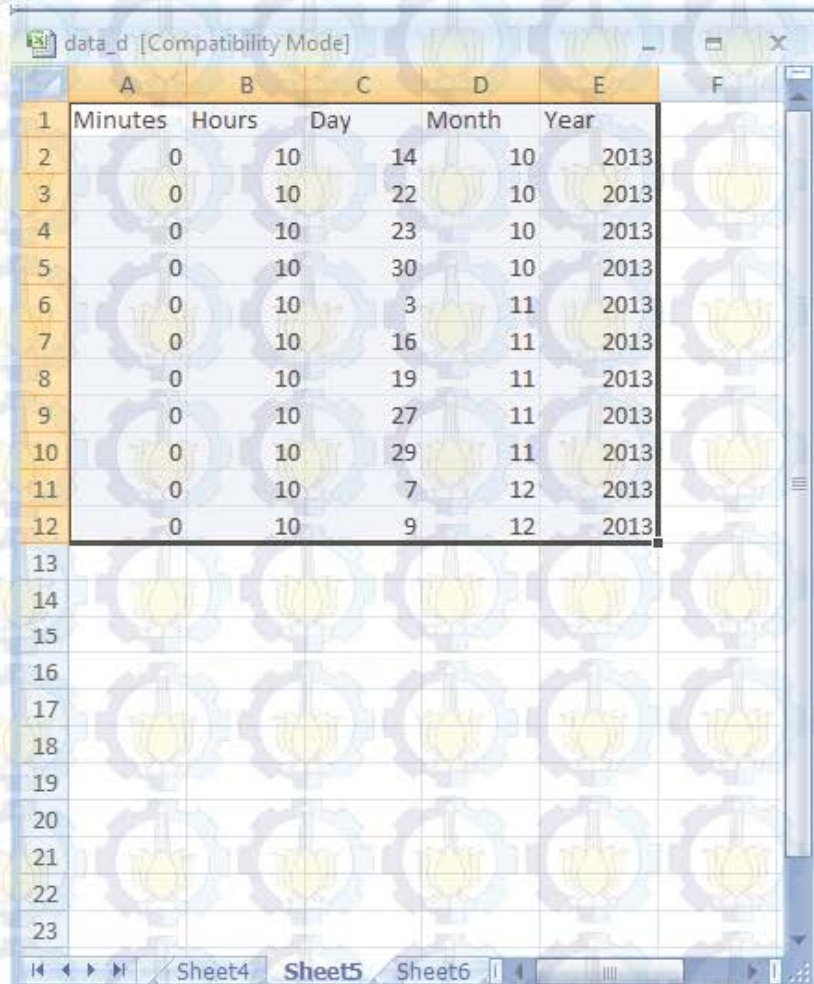


Sheet 4: Lama Perjalanan Berangkat atau Kembali dari TBBM *Customer* untuk  
Volume Permintaan  $\leq 22.000$  KI



	A	B	C	D	E	F
1	22					
2	22					
3	16,2					
4	22					
5	22					
6	16,2					
7	22					
8	16,2					
9	22					
10	22					
11	22					
12	22					
13	22					
14	16,2					
15	22					
16						
17						
18						
19						
20						
21						
22						
23						

Sheet 5: Waktu Permintaan untuk 22.000 KI-25.000 KI



	A	B	C	D	E	F
1	Minutes	Hours	Day	Month	Year	
2	0	10	14	10	2013	
3	0	10	22	10	2013	
4	0	10	23	10	2013	
5	0	10	30	10	2013	
6	0	10	3	11	2013	
7	0	10	16	11	2013	
8	0	10	19	11	2013	
9	0	10	27	11	2013	
10	0	10	29	11	2013	
11	0	10	7	12	2013	
12	0	10	9	12	2013	
13						
14						
15						
16						
17						
18						
19						
20						
21						
22						
23						

Sheet 7: Lama *Loading* atau *Unloading* untuk Volume Permintaan 22.000 KI-  
25.000 KI

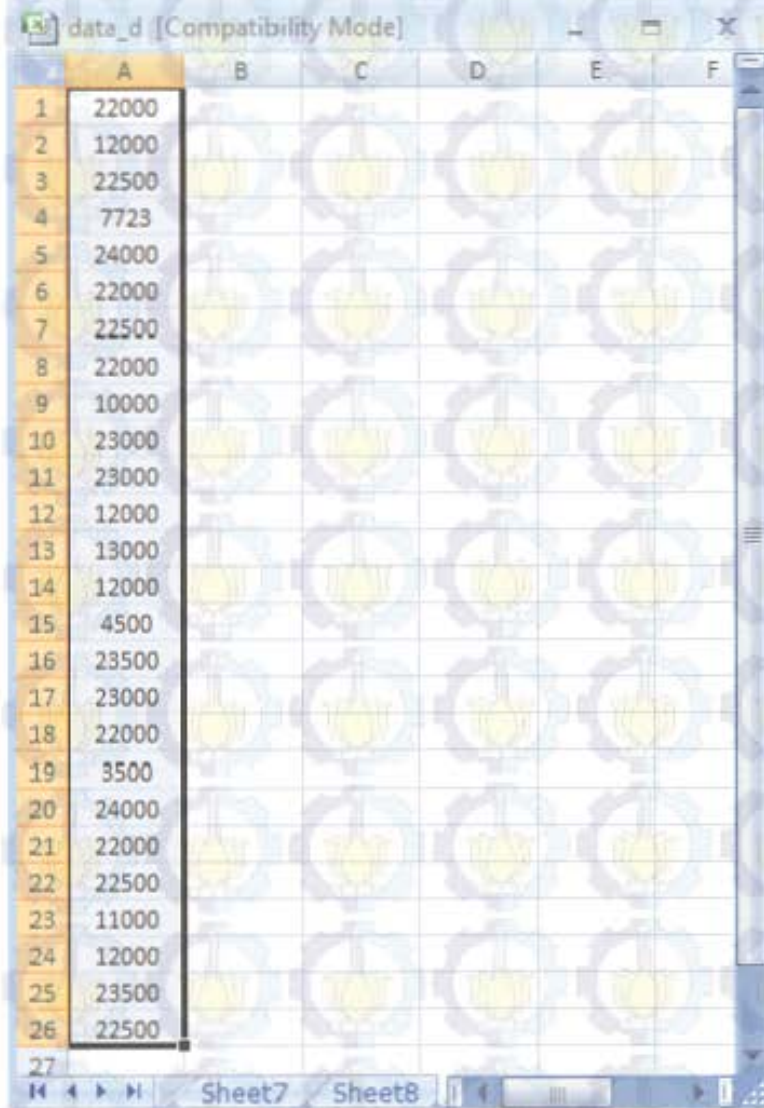
	A	B	C	D	E	F
1	45					
2	48					
3	45					
4	46					
5	46					
6	47					
7	46					
8	48					
9	45					
10	47					
11	45					
12						
13						
14						
15						
16						
17						
18						
19						
20						
21						
22						
23						



Sheet 8: Lama Perjalanan Berangkat atau Kembali dari TBBM *Customer* untuk  
Volume Permintaan 22.000 KI-25.000 KI

	A	B	C	D	E	F
1	22					
2	16,2					
3	16,2					
4	16,2					
5	22					
6	16,2					
7	16,2					
8	16,2					
9	22					
10	16,2					
11	22					
12						
13						
14						
15						
16						
17						
18						
19						
20						
21						
22						
23						

Sheet 10: Volume dari 26 Permintaan untuk Model kedua



	A	B	C	D	E	F
1	22000					
2	12000					
3	22500					
4	7723					
5	24000					
6	22000					
7	22500					
8	22000					
9	10000					
10	23000					
11	23000					
12	12000					
13	13000					
14	12000					
15	4500					
16	23500					
17	23000					
18	22000					
19	3500					
20	24000					
21	22000					
22	22500					
23	11000					
24	12000					
25	23500					
26	22500					
27						

## Lampiran 5

### Listing Program Fungsi Supplychain1to2b

```
function supplychain1to2b
//Mendefinisikan m,md dan e
m=3;
md=15;
e=;%inf;

//input tanggal permintaan melalui file xls
datacontoh = uigetfile('*.xls','Title=Choose xls for date of demand');
if datacontoh == "
    disp('You exit from this process');
    return;
end

//Input data dari file xls
Sheets = readxls(datacontoh);

//Data tanggal permintaan kurang dari atau sama dengan 22.000 Kl
s1= Sheets(1);
tanggal_permintaanb=s1.value;

//Merubah data tanggal, bulan dan tahun ke bentuk integer
data_Yb=zeros(md,1);
for i=1:md
    data_Yb(i,1)=date2intc(tanggal_permintaanb(i+1,1),tanggal_permintaanb(i+1,2),tanggal_permintaanb(i+1,3),tanggal_permintaanb(i+1,4),tanggal_permintaanb(i+1,5),2013);
end

//Data lama proses loading atau unloading
s2= Sheets(3);
lo=s2.value;
lub=lo*60;
load_unloadb=zeros(md,1);
for i=1:md
    load_unloadb(i,1)=lub(i,1);
end

//Data lama berangkat atau kembali
s3 = Sheets(4);
be=s3.value;
berangkat_kembalib=be*60;

pb=load_unloadb+berangkat_kembalib+load_unloadb;
pdb = berangkat_kembalib;

//Mendefinisikan matriks Ac,Bc dan Cc utk volume tdk melebihi 22.000 kl yg dinotasikan dg Ab, Bb dan Cb
Ab = zeros(2,2*md)
for i = 4:md
    Ab(:,2*i-1:2*i) = [e pdb(i-m);e maxplusotimes(pb(i),pdb(i-m))];
end

Bb = zeros(2,md)
for i = 1:md
    Bb(:,i) = [e;pb(i)];
end
```



```

Cb = [e 0];

//Membentuk matriks Yb
Yb = zeros(md/m,m);
for i = 1:md/m
    for j = 0:m-1
        Yb(i,j+1) = data_Yb(1+(3*(i-1)+j));
    end
end
Yb1=Yb(:,1);
Yb2=Yb(:,2);
Yb3=Yb(:,3);

//Membentuk matriks Hb1,Hb2,dan Hb3
Hb1 = maxpluszeros(md/m,md/m);
for i = 1:md/m
    for j=1:md/m
        if i==j then
            Hb1(i,j)=maxplusotimes(Cb,Bb(:,1+(i-1)*m));

        elseif i==j+1 then
            Hb1(i,j)=maxplusotimes(Cb, maxplusotimes(Ab(:,6*i-5:6*i-4),Bb(:,1+(i-2)*m)));

        elseif i==j+2 then
            Hb1(i,j) = maxplusotimes(maxplusotimes(Cb,Ab(:,6*i-5:6*i-4)), maxplusotimes(Ab(:,6*i-11:6*i-10),Bb(:,1+(i-3)*m)));

        elseif i==j+3 then
            Hb1(i,j) = maxplusotimes(Cb, maxplusotimes(maxplusotimes(Ab(:,6*i-5:6*i-4),Ab(:,6*i-11:6*i-10)), maxplusotimes(Ab(:,6*i-17:6*i-16),Bb(:,1+(i-4)*m))))

        elseif i==j+4 then
            Hb1(i,j) = maxplusotimes(maxplusotimes(Cb,Ab(:,6*i-5:6*i-4)), maxplusotimes(maxplusotimes(Ab(:,6*i-11:6*i-10),Ab(:,6*i-17:6*i-16)), maxplusotimes(Ab(:,6*i-23:6*i-22),Bb(:,1+(i-5)*m))))

        end
    end
end

Hb2 = maxpluszeros(md/m,md/m);
for i = 1:md/m
    for j=1:md/m
        if i==j then
            Hb2(i,j)=maxplusotimes(Cb,Bb(:,2+(i-1)*m));

        elseif i==j+1 then
            Hb2(i,j)=maxplusotimes(Cb, maxplusotimes(Ab(:,6*i-3:6*i-2),Bb(:,2+(i-2)*m)));

        elseif i==j+2 then
            Hb2(i,j) = maxplusotimes(maxplusotimes(Cb,Ab(:,6*i-3:6*i-2)), maxplusotimes(Ab(:,6*i-9:6*i-8),Bb(:,2+(i-3)*m)));

        elseif i==j+3 then
            Hb2(i,j) = maxplusotimes(Cb, maxplusotimes(maxplusotimes(Ab(:,6*i-3:6*i-2),Ab(:,6*i-9:6*i-8)), maxplusotimes(Ab(:,6*i-15:6*i-14),Bb(:,2+(i-4)*m))))

        elseif i==j+4 then
            Hb2(i,j) = maxplusotimes(maxplusotimes(Cb,Ab(:,6*i-3:6*i-2)), maxplusotimes(maxplusotimes(Ab(:,6*i-9:6*i-8),Ab(:,6*i-15:6*i-14)), maxplusotimes(Ab(:,6*i-21:6*i-20),Bb(:,2+(i-5)*m))))

    end
end

```

```

end
end
end

Hb3 = maxpluszeros(md/m,md/m);
for i = 1:md/m
    for j = 1:md/m
        if i==j then
            Hb3(i,j)=maxplusotimes(Cb,Bb(:,3+(i-1)*m));

        elseif i==j+1 then
            Hb3(i,j)=maxplusotimes(Cb, maxplusotimes(Ab(:,6*i-1:6*i),Bb(:,3+(i-2)*m)));

        elseif i==j+2 then
            Hb3(i,j) = maxplusotimes(maxplusotimes(Cb,Ab(:,6*i-1:6*i)), maxplusotimes(Ab(:,6*i-7:6*i-6),Bb(:,3+(i-3)*m)));

        elseif i==j+3 then
            Hb3(i,j) = maxplusotimes(Cb, maxplusotimes(maxplusotimes(Ab(:,6*i-1:6*i),Ab(:,6*i-7:6*i-6)), maxplusotimes(Ab(:,6*i-13:6*i-12),Bb(:,3+(i-4)*m))))

        elseif i==j+4 then
            Hb3(i,j) = maxplusotimes(maxplusotimes(Cb,Ab(:,6*i-1:6*i)), maxplusotimes(maxplusotimes(Ab(:,6*i-7:6*i-6),Ab(:,6*i-13:6*i-12)), maxplusotimes(Ab(:,6*i-19:6*i-18),Bb(:,3+(i-5)*m))))

        end
    end
end

//Menghitung nilai Ub
U1=maxplusinsol(Hb1,Yb1);
U2=maxplusinsol(Hb2,Yb2);
U3=maxplusinsol(Hb3,Yb3);
Ub=[U1 U2 U3];
Ubtemp = matrix(Ub',1,md);
Ubakhir = Ubtemp(1,md);

//Menentukan waktu berangkatb
berangkatb=Ubakhir+load_unloadb;

//Menentukan waktu produk diterimab
terimab=berangkatb+berangkat_kembalib+load_unloadb;

//Menentukan waktu kembali dari pengirimanb
pulangb=terimab+berangkat_kembalib

//Menentukan Selisih tanggal permintaan dengan sampainya produk di TBBM customer
selisihb=data_Yb-terimab

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
////

//Mendefinisikan m dan md untuk volume 22.000 KI-25.000 KI yang dinotasikan dengan z dan zd
z=2;
zd=11;

//Data tanggal permintaan 22.000 KI-25.000 KI
s4= Sheets(5);
tanggal_permintaanc=s4.value;

//Merubah data tanggal, bulan dan tahun ke bentuk integer
data_Yc=zeros(zd,1);

```



```

for i=1:zd
data_Yc(i,1)=date2intc(tanggal_permintaanc(i+1,1),tanggal_permintaanc(i+1,2),tanggal_permintaanc(i+1,3),tanggal_permintaanc(i+1,4),tanggal_permintaanc(i+1,5),2013);
end

//Data lama proses loading atau unloading
s5= Sheets(7);
lox=s5.value;
load_unloadc=zeros(zd,1);
for i=1:zd
load_unloadc(i,1)=lox(i,1)*60;
end

//Data lama berangkat atau kembali
s6 = Sheets(8);
bec=s6.value;
berangkat_kembalic=zeros(zd,1);
for i=1:zd
berangkat_kembalic(i,1)=bec(i,1)*60;
end
pc=load_unloadc+berangkat_kembalic+load_unloadc;
pdc = berangkat_kembalic;

//Mendefinisikan matrik Ac dan Bc
Ac = zeros(2,2*zd)
for i = 4:zd
Ac(:,2*i-1:2*i)= [e pdc(i-z);e maxplusotimes(pc(i),pdc(i-z))];
end

Bc = zeros(2,zd)
for i = 1:zd
Bc(:,i)= [e;pc(i)];
end

//Membentuk matriks Yc1 dan Yc2
Yc1 = zeros(6,1);
for i = 0:5
Yc1(i+1,1) = data_Yc(1+z*i);
end
Yc2 = zeros(5,1);
for i = 0:4
Yc2(i+1,1) = data_Yc(1+z*i+1);
end

//Membentuk matriks Hc1 dan Hc2
Hc1 = maxpluszeros(ceil(zd/z),ceil(zd/z));
for i = 1:ceil(zd/z)
for j=1:ceil(zd/z)
if i==j then
Hc1(i,j)=maxplusotimes(Cb,Bc(:,2*i-1));
elseif i==j+1 then
Hc1(i,j)=maxplusotimes(Cb, maxplusotimes(Ac(:,4*i-3:4*i-2),Bc(:,2*i-3)));
elseif i==j+2 then
Hc1(i,j) = maxplusotimes(maxplusotimes(Cb,Ac(:,4*i-3:4*i-2)), maxplusotimes(Ac(:,4*i-7:4*i-6),Bc(:,2*i-5)));
elseif i==j+3 then
Hc1(i,j) = maxplusotimes(Cb, maxplusotimes(maxplusotimes(Ac(:,4*i-3:4*i-2),Ac(:,4*i-7:4*i-6)), maxplusotimes(Ac(:,4*i-11:4*i-10),Bc(:,2*i-7))))

```



```

        elseif i==j+4 then
            Hc1(i,j) = maxplusotimes(maxplusotimes(Cb,Ac(:,4*i-3:4*i-2)),
maxplusotimes(maxplusotimes(Ac(:,4*i-7:4*i-6),Ac(:,4*i-11:4*i-10)), maxplusotimes(Ac(:,4*i-15:4*i-
14),Bc(:,2*i-9))))

        elseif i == j+5 then
            Hc1(i,j) = maxplusotimes(maxplusotimes(Cb,maxplusotimes(Ac(:,4*i-3:4*i-2),Ac(:,4*i-7:4*i-6)),
maxplusotimes(maxplusotimes(Ac(:,4*i-11:4*i-10),Ac(:,4*i-19:4*i-18)), maxplusotimes(Ac(:,4*i-23:4*i-
22),Bc(:,2*i-11))))
        end
    end
end

Hc2 = maxpluszeros(fix(zd/z),fix(zd/z));
for i = 1:fix(zd/z)
    for j=1:fix(zd/z)
        if i==j then
            Hc2(i,j)=maxplusotimes(Cb,Bc(:,2*i));

        elseif i==j+1 then
            Hc2(i,j)=maxplusotimes(Cb, maxplusotimes(Ac(:,4*i-3:4*i-2),Bc(:,2*i-2)));

        elseif i==j+2 then
            Hc2(i,j) = maxplusotimes(maxplusotimes(Cb,Ac(:,4*i-3:4*i-2)), maxplusotimes(Ac(:,4*i-7:4*i-
6),Bc(:,2*i-3)));

        elseif i==j+3 then
            Hc2(i,j) = maxplusotimes(Cb, maxplusotimes(maxplusotimes(Ac(:,4*i-3:4*i-2),Ac(:,4*i-7:4*i-6)),
maxplusotimes(Ac(:,4*i-11:4*i-10),Bc(:,2*i-4))))

        elseif i==j+4 then
            Hc2(i,j) = maxplusotimes(maxplusotimes(Cb,Ac(:,4*i-3:4*i-2)),
maxplusotimes(maxplusotimes(Ac(:,4*i-7:4*i-6),Ac(:,4*i-11:4*i-10)), maxplusotimes(Ac(:,4*i-15:4*i-
14),Bc(:,2*i-5))))

        end
    end
end

//Menghitung nilai Uc
Uc1=maxplusinsol(Hc1,Yc1);
Uc0=maxplusinsol(Hc2,Yc2);
Uc2=zeros(ceil(zd/z),1);
for i=1:ceil(zd/z)-1
    Uc2(i,1)=Uc0(i,1);
end
for i=ceil(zd/z)
    Uc2(i,1)=0;
end
Uc=[Uc1 Uc2];
Uctemp = matrix(Uc',1,zd+1);
Uca = Uctemp(1:zd+1)';
Ucakhir=Uca(1:zd,1);

//Menentukan waktu berangkat
berangkat=Ucakhir+load_unloadc;

//Menentukan waktu produk diterimac
terimac=berangkat+berangkat_kembalic+load_unloadc;

//Menentukan waktu kembali dari pengirimanc
pulangc=terimac+berangkat_kembalic

```

*//Menentukan Selisih tanggal permintaan dengan sampainya produk di TBBM customer*  
selisihc=data\_Yc-terimac

*//Menentukan urutan kapal awal*

```
kapalb=zeros(md,1);
for i=1:md
    for j=0:md/m
        kapalb(i+j*3,1)=i
    end
end
kapalb=kapalb(1:md,1);
kapalc=zeros(zd,1);
for i=1:zd
    for j=0:zd/z
        kapalc(i+j*2,1)=i+3
    end
end
kapalc=kapalc(1:zd,1);
```

*//Menggabungkan permintaan kurang dari atau sama dengan 22.000 KI dengan 22.000 KI-25.000 KI*

```
t_p=[data_Yb;data_Yc];
t_l=[Ubakhir;Ucakhir];
t_b=[berangkatb;berangkatc];
t_t=[terimab;terimac];
t_k=[pulangb;pulangc];
t_permintaan=lex_sort(t_p);
t_loading=lex_sort(t_l);
t_berangkat=lex_sort(t_b);
t_terima=lex_sort(t_t);
t_pulang=lex_sort(t_k);

kapal=[kapalb(1:2,1);kapalc(1,1);kapalb(3,1);kapalc(2,1);kapalb(4,1);kapalc(3,1);kapalb(5:6,1);kapalc(4:5,1);
kapalb(7:10,1);kapalc(6:7,1);kapalb(11:12,1);kapalc(8,1);kapalb(13,1);kapalc(9,1);kapalb(14:15,1);kapalc(10:11,1)];
```

```
vo=[load_unloadb(1:2,1);load_unloadc(1,1);load_unloadb(3,1);load_unloadc(2,1);load_unloadb(4,1);load_unloadc(3,1);load_unloadb(5:6,1);load_unloadc(4:5,1);load_unloadb(7:10,1);load_unloadc(6:7,1);load_unloadb(11:12,1);load_unloadc(8,1);load_unloadb(13,1);load_unloadc(9,1);load_unloadb(14:15,1);load_unloadc(10:11,1)];
volume=vo/60*500;
```

*//Merubah waktu loadingb, berangkatb, produk diterimab dan kembali dari pengirimanb ke bentuk menit, jam, tanggal, bulan dan tahun*

```
tanggal_permintaan=zeros(md+zd,5);
for i=1:md+zd
    [tanggal_permintaan(i,1),tanggal_permintaan(i,2),tanggal_permintaan(i,3),tanggal_permintaan(i,4),tanggal_permintaan(i,5)]=int2dateetc(t_permintaan(i,1),2013);
end

tanggal_loading=zeros(md+zd,5);
for i=1:md+zd
    [tanggal_loading(i,1),tanggal_loading(i,2),tanggal_loading(i,3),tanggal_loading(i,4),tanggal_loading(i,5)]=int2dateetc(t_loading(i,1),2013);
end

tanggal_berangkat=zeros(md+zd,5);
for i=1:md+zd
    [tanggal_berangkat(i,1),tanggal_berangkat(i,2),tanggal_berangkat(i,3),tanggal_berangkat(i,4),tanggal_berangkat(i,5)]=int2dateetc(t_berangkat(i,1),2013);
end
```



```

tanggal_terima=zeros(md+zd,5);
for i= 1:md+zd

[tanggal_terima(i,1),tanggal_terima(i,2),tanggal_terima(i,3),tanggal_terima(i,4),tanggal_terima(i,5)]=int2date
etc(t_terima(i,1),2013);
end

tanggal_pulang=zeros(md+zd,5);
for i= 1:md+zd

[tanggal_pulang(i,1),tanggal_pulang(i,2),tanggal_pulang(i,3),tanggal_pulang(i,4),tanggal_pulang(i,5)]=int2da
teetc(t_pulang(i,1),2013);
end

//Menampilkan hasil di tabel
number=[1:md+zd]';
number=string(number);
volume=string(volume);
kapal=string(kapal);
jadwal_load=[];
jadwal_berangkat=[];
jadwal_terima=[];
jadwal_permintaan=[];
jadwal_kembali=[];
for i=1:md+zd

jadwal_load=[jadwal_load;date2stretc(tanggal_loading(i,3),tanggal_loading(i,4),tanggal_loading(i,5),tanggal
_loading(i,2),tanggal_loading(i,1))];

jadwal_berangkat=[jadwal_berangkat;date2stretc(tanggal_berangkat(i,3),tanggal_berangkat(i,4),tanggal_bera
ngkat(i,5),tanggal_berangkat(i,2),tanggal_berangkat(i,1))];

jadwal_terima=[jadwal_terima;date2stretc(tanggal_terima(i,3),tanggal_terima(i,4),tanggal_terima(i,5),tanggal
_terima(i,2),tanggal_terima(i,1))];

jadwal_permintaan=[jadwal_permintaan;date2stretc(tanggal_permintaan(i,3),tanggal_permintaan(i,4),tanggal
_permintaan(i,5),tanggal_permintaan(i,2),tanggal_permintaan(i,1))];

jadwal_kembali=[jadwal_kembali;date2stretc(tanggal_pulang(i,3),tanggal_pulang(i,4),tanggal_pulang(i,5),ta
nggal_pulang(i,2),tanggal_pulang(i,1))];
end

data=['No.','Volume','Kapal','Jadwal Loading','Jadwal Berangkat','Jadwal Terima','Tanggal
Permintaan','Jadwal
Kembali'];[number,volume,kapal,jadwal_load,jadwal_berangkat,jadwal_terima,jadwal_permintaan,jadwal_ke
mbali]];

f = gcf();
f.position=[10,10,900,400]
clf
as = f.axes_size; // [width height]
ut = uicontrol("style","table",...
"string",data,...
"position",[5 as(2)-300 650 287],...// => @top left corner of figure
"tooltipstring","Data from majors towns")
delete(ut);
ut = uicontrol("style","table",...
"string",data,...
"position",[5 as(2)-300 650 287],...// => @top left corner of figure
"tooltipstring","Data from majors towns")
Endfunction

```